LA RIVISTA DI ATARI®

LA PUBBLICAZIONE
JACKSON PER GLI UTENTI
DEI SISTEMI ATARI

Anno 1 n° 3 Maggio/Giugno 1987 - L. 5.000 Sped. in Abb. Post. Gr. IV/70%

USO E ABUSO
DI UN FLOPPY DISK

8 BIT:
COMPUTEREYES

SPECIALE ST
TELECOMUNICAZIONI

TEORIA:
I SEGRETI DEL GEM

PROVE SW:
DB MAN,
AEGIS ANIMATOR

Foto di copertina
Carlo Scillieri

1987

EDITORIALE

L'ATTESA HA ORMAI RISCALDATO GLI ANIMI, E NON SI CONTANO LE TELEFONATE E LE LETTERE CHE SOMMERGONO LA REDAZIONE ALLA RICERCA DI INFORMAZIONI SULLE NOVITÀ IN ARRIVO. LE NUOVE MACCHINE PRESENTATE DA ATARI ITALIA AL SIOA DI BOLOGNA HANNO SCATENATO L'ENTUSIASMO DEGLI ATARIANI E L'INVIDIA DELLA CONCORRENZA. L'APERTURA DI ATARI AL MONDO MS-DOS, LO STANDARD INDUSTRIALE NEL CAMPO DEI PERSONAL COMPUTER PER L'OFFICE AUTOMATION, È SICURAMENTE UN FATTO IMPORTANTE, CONFERMATO DALL'INTERESSE CHE GLI OPERATORI DEL SETTORE E IL PUBBLICO HANNO MOSTRATO DOPO GLI ANNUNCI DELL'ATARI PC. CHI TEME CHE LA RIVISTA DI ATARI SI TRASFORMI IN UN PERIODICO PER PC IBM E COMPATIBILI SI TRANQUILLIZZI: LO SPAZIO DEDICATO AL NUOVO MONDO IN CUI ATARI È ENTRATA DI PREPOTENZA SARÀ LIMITATO ALL'ESSENZIALE. LARGO AGLI ST, DUNQUE, DA MEZZO MEGA IN SU!

Diego Biasi

la rivista di ATARI
DIRETTORE RESPONSABILE
Gianpietro Zanga
DIRETTORE
Diego Biasi
COLLABORATORI
Emanuele Bergamini, Marino Bonetti, Andrea Borroni, Dario Bressanini, Roberto Cazzaro, Paolo Galvani, Daniele Guarino, Maurizio Miccoli, Luca Mantegazza, Giovanni Menna, Mauro Pavone, Matteo Prinetti, Mauro Scaioni
ART DIRECTOR
Giovanna Ghezzi
FOTOGRAFIE
Paolo Galvani, Maurizio Miccoli, Carlo Scillieri
REDAZIONE
BY BYTE s.n.c.
Servizi per la comunicazione
Corso di P.ta Romana, 1
20122 Milano
tel. 879992-870824

DIVISIONE PUBBLICITÀ
via Pola, 9 - 20124 Milano
tel. 6948.1
telex 316213 REINA I
33436 GEJIT I
CONCESSIONARIA DI ROMA
Francesca Juvara
UNION MEDIA Srl
via G.B. Martini, 13
00198 Roma
tel. (06) 8119803-4
telex 630206 UNION I
UFFICIO ABBONAMENTI
via Rosellini, 12 - 20124 Milano
tel. (02) 6880951
prezzo della rivista L. 5.000
prezzo per l'estero L. 10.000
numero arretrato L. 10.000
Abbonamento annuo L. 24.000
per l'Estero L. 48.000
I versamenti vanno indirizzati a:
Gruppo Editoriale Jackson S.p.A.
via Rosellini, 12 - 20124 Milano
mediante emissione di assegno bancario, o cartolina vaglia o utilizzando il c/c postale
n. 11666203

DIREZIONE E AMMINISTRAZIONE
via Rosellini, 12 - 20124 Milano
tel. 680368-680054-6880951.2.3.4.5
telex 333436 GEJ ITI
SEDE LEGALE:
Via G. Pozzone, 5
20121 Milano
CONSOCIATE ESTERE
U.S.A. - GEJ Publishing Group, Inc. 125 University Avenue Palo Alto, CA 94301 - tel. (415) 3225885
U.K. - GEJ Publishing Ltd 18 Oxford Street London WIR 1AJ
tel. (01) 4392931
telex (051) 21248
SPAGNA - Jackson Hispania s.a. Plaza Republica del Ecuador, 2
28016 Madrid
tel. 00/34/1/4579424
telex (052) 49371 ELOCE
Fotocomposizione:
GDB - Milano
Stampa: GRAFIKA 78
Pioltello - Milano
Concessionario esclusivo per la diffusione in Italia:
SODIP - Via Zuretti, 25
20125 Milano
Spedizione in abbonamento postale gruppo IV/70%
Autorizzazione alla pubblicazione Tribunale di Milano
numero 19 del 16/1/1987

Vi scrivo per ringraziarvi di aver realizzato "La Rivista di Atari", indispensabile strumento per la crescita e la diffusione di un personal computer eccezionale come il nostro ST.
Vorrei chiedervi se pensate di realizzare una rubrica come "Bitborsa" di Bit, indispensabile per permettere all'Atariano di contattare altri colleghi per scambio di informazioni su hardware e software.
Per quanto riguarda il questionario di pagina 35 del n. 1, vi rispondo (non utilizzo il tagliando per non rovinare la Rivista) che il giornale è soddisfacente, anche se a mio parere manca una rubrica prettamente hardware per gli "smanettoni" come me.
Complimenti ed auguri per la rivista che spero diventi presto mensile!!

Fabio Micheloni
Roma

Innanzitutto grazie per gli elogi, che ci spronano a fare sempre meglio.
E grazie anche per i numerosi suggerimenti: miglioreremo con la vostra collaborazione. Venendo alla tua domanda non possiamo che risponderti: mandate i vostri annunci, e appena ne avremo un numero sufficiente organizzeremo una rubrica simile a "Bitborsa". Dubitiamo però che ne arriveranno molti: chi ha un Atari se lo tiene!
Scherzi a parte, il tuo annuncio è già stato messo in lista per una prossima pubblicazione che avverrà appena possibile.
Prevedere se "La Rivista di Atari" diverrà mensile è un po' presto, però vista l'entusiastica risposta che le avete riservato studieremo la proposta...

Spettabile Redazione, innanzitutto vorrei farvi i complimenti per la vostra Rivista, che mi soddisfa ampiamente. Ma veniamo al dunque.
Sono un utente Atari possessore di un 130 XE da più di un anno. In questo periodo ho imparato ad usarlo e mi diletto a scrivere programmi, ma, contemporaneamente, mi sono messo alla ricerca di negozi che vendessero delle utility per il mio computer. Devo però ammettere con delusione che non sono riuscito a trovarne. A questo proposito vi pregherei di indicarmi alcuni rivenditori della mia zona.
Dato che il mio computer utilizza come memoria esterna un disk drive, vorrei chiedervi se esistono o sono in programma libri accompagnati da disco.
Distinti saluti.

Francesco Rizzi
Pasian di Prato (UD)

I rivenditori Atari della tua zona li puoi trovare nelle pagine in fondo alla Rivista. Per tua comodità riportiamo gli indirizzi dei negozi presenti in Udine:
– Il giocattolo 2, Via Mercato Vecchio 29
– Mofert snc, Via Europa Unita 41
Per quanto riguarda libri con allegati dischi per l'Atari 130XE, per ora non è previsto nulla di simile. Se dovessimo registrare qualche novità, ne daremo notizia sulle pagine della Rivista.

Accidenti, vi siete decisi finalmente a fare una rivista tuttAtari per i possessori di questi non troppo fortunati computer qui in Italia. La Rivista in sè stessa non è affatto male (come impostazione assomiglia un po' al vecchio e glorioso Videogiochi & Computer) anche se, come avete detto voi, andrebbero ampliate le pagine riguardanti le prove dei videogames e i listati.
Per le prove HW, perché invece di provare un solo prodotto non fate il confronto con lo stesso accessorio ma per altri computer?
Spero anche che vogliate inserire uno spazio per le gare dei computer da casa (XE, XL, ST e, perché no, 2600VCS) che in un giornale come il vostro a mio avviso non possono mancare.
Mi è piaciuto moltissimo il servizio del club di Monza dove non si copiano programmi, ma si impara a programmare sul serio e ad usare il computer come mezzo sociale per conoscersi e scambiare idee. Purtroppo io abito in Toscana, per cui penso che non avrò mai la possibilità di partecipare ad una riunione di "Artigiani".
In questi ultimi tempi non ho trovato praticamente nessun programma nuovo in giro, anche se case distributrici di soft (Lago e Mastertronic) avevano preannunciato un'immissione di nuovi programmi sul mercato, cosa però che a quanto vedo tarda ad accadere. Perché questi ritardi?
Con questo concludo la mia missiva che vuol essere un elogio al vostro coraggioso lavoro ed uno sprone a tutti i possessori di Atari.
Ciao e a presto.

Damiano Masi
Lamporecchio (PT)

Ehi!!! Andiamoci piano nel dire che le macchine Atari hanno avuto poca fortuna in Italia. Non abbiamo sottomano i dati precisi di vendita degli ultimi mesi, ma ti possiamo garantire che le cose stanno andando a gonfie vele. La prova è data dall'esistenza di questa rivista e dall'impressionante numero di lettere che riceviamo ogni giorno!
Terremo sicuramente in considerazione la tua proposta di comparare lo stesso prodotto per diversi computer. Ne parleremo in Redazione e decideremo cosa fare. Al mo-

mento non è proponibile invece l'idea di includere una gara per home computer: lo spazio è sempre tiranno e gli argomenti certo non mancano. In ogni caso anche per questo aspettiamo di vedere cosa ne pensano gli altri lettori.
Le novità software invece sono arrivate, e te ne sarai reso conto vedendo l'inserzione della Mastertronic sullo scorso numero che riportava alcune delle ultime novità. Altre ce ne saranno anche in futuro.

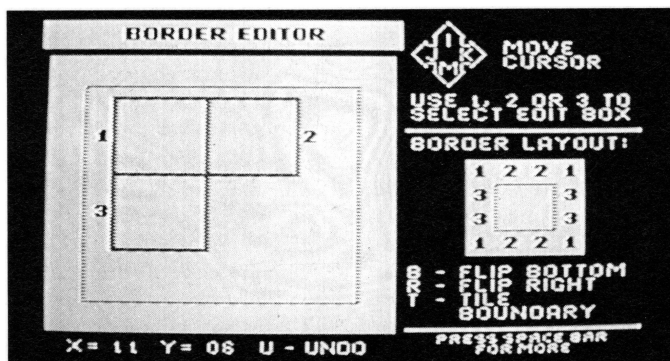
Dopo aver compilato il questionario, scrivo due righe per dire che il 520 ST l'ho appena comprato e i programmi li ho appena fatti arrivare dalla Francia perché sulle riviste italiane non ho trovato inserzioni di rivenditori con l'elenco dei programmi che forniscono (e non ultima cosa in Francia si compra Hors Taxe: in Italia l'IVA si paga!).
A parte questo vorrei sapere come posso fare per abbonarmi alla Vostra Rivista, l'unica in italiano che parli dell'Atari (tenete presente che lavoro undici mesi su dodici) e dove posso rivolgermi per i programmi. In Francia sono comodi, ma poi c'è la noia di tradurre tutto in italiano.
Chiedo scusa se vi ho disturbato. Distinti saluti.
P.S.: Mi interessano in particolare programmi di contabilità.

Luciano Furlan
Constantine (Algeria)

I lettori non ci disturbano mai, anzi. E poi ci fa piacere sapere che abbiamo amici addirittura in Algeria.
Per abbonarsi deve effettuare un pagamento tramite assegno bancario, o vaglia, o versamento sul c/c postale n. 11666203 intestato a Gruppo Editoriale Jackson SpA, Via Rosellini 12 - 20124 Milano, specificando la richiesta di abbonamento alla "Rivista di Atari". L'importo è di 24.000 lire (se desidera ricevere la Rivista in Italia) oppure di 48.000 lire (abbonamento annuo per l'estero).
Per i programmi le consigliamo di rivolgersi alla Hard & Soft (Servizio Clienti - Via Sem Benelli 8 - 20151 Milano) che ha un vastissimo catalogo di software per Atari e che effettua anche vendita per corrispondenza, o alla Mastertronic (servizio Vendita per Corrispondenza) di Casciago (Varese).
Sperando di esserle stati di aiuto la salutiamo e la ringraziamo per l'attenzione.

Spettabile Redazione de "La Rivista di Atari", ho letto il primo numero del giornale e mi è piaciuto moltissimo, soprattutto per la varietà delle rubriche, dalle News alle prove hardware, dai listati alle prove software.
Vi ho scritto questa lettera per farvi alcune domande a cui spero mi possiate dare una risposta (possiedo un 800 XL con registratore):
1 - In una delle prove software per ST c'è scritto che i giochi Silent Service e Leader Board sono stati prodotti anche per la serie XL/XE: esistono in Italia le versioni su cartuccia e su cassetta per l'XL? Se sì, dove le posso trovare a Milano?
2 - Al Tempio del Computer di Milano e nei punti vendita Atari posso trovare alcuni giochi su cassetta o cartuccia per l'Atari XL?
3 - Dove sono a Milano i negozi Bit Shop Primavera?
Attendo cortesemente una vostra risposta non prima di avervi rinnovato i miei complimenti per la fantastica rivista.
Cordiali saluti.

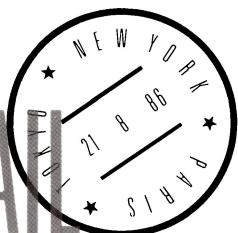
Andrea Bottos
Milano

Eccoti tre risposte lampo per i tuoi quesiti.
I giochi Silent Service e Leader Board sono disponibili per i computer della serie XL solamente su disco. Per averli prova a rivolgerti ai rivenditori Atari di Milano che compaiono in fondo alla rivista.
Risposta positiva per il reperimento dei programmi su cassetta e cartuccia nei negozi di Milano: non dovresti avere alcun problema.
La Bit Shop Computers (ex Bit Shop Primavera) è presente a Milano in Via Carlo Farini, 82 - Tel. 6880429 - 6882171.

Una schermata di "The Print Shop"

Spettabile Redazione, ho aspettato con ansia l'uscita del secondo numero della vostra egregia rivista, notando subito la risposta alla lettera che avevo precedentemente spedito, il che mi ha fatto molto piacere e ridere. La domanda, o meglio le domande, che vi pongo questa volta riguardano l'ormai attesissimo Blitter. A tale proposito vorrei sapere:
– come funziona
– come si installa (e da chi)
– quando lo metteranno in commercio.
Giungendo al termine di questa lettera (detto fra noi, perché qualche volta non mettete in copertina una bella ragazza?!) e sperando nel buon esito della vostra rivista, finisco porgendovi i miei migliori auguri.

Riccardo Loffi
Cremona

Caro Riccardo, non ti sarai mica messo in testa di diventare ricco scrivendo alla "Rivista di Atari"? A partire da questo numero non paghiamo più le lettere!
Torniamo a un po' di serietà. L'atteso Blitter si fa ancora attendere. L'Atari Italia, interpellata in proposito, ha detto che non è possibile preventivare il tempo di attesa che ancora occorrerà prima di poter commercializzare nel nostro paese questo chip grafico.
Di conseguenza non sappiamo dirti nemmeno se l'installazione verrà fatta presso i centri di assistenza Atari o da qualche altra parte. Non appena avremo notizie le pubblicheremo.

Spettabile Redazione, vi scrivo innanzitutto per complimentarmi per la bella Rivista. La trovo eccezionale, buona sotto tutti i punti di vista, e soprattutto rende giustizia a noi Atariani, o Ataristi (a proposito come si dice?), che vedevamo relegato il nostro home (leggi 400-800-XL-XE) al ruolo di Cenerentola degli "8 bit", vedendo in edicola tante riviste e prodotti per i computer più in voga (Commodore, Sinclair, Sharp, sic!). Vedevamo sui volti dei nostri amici-antagonisti sorrisetti maliziosi e commiserevoli, come chi cerca di consolare un amico che ha avuto la sfortuna di fare un acquisto a dir poco sbagliato e che segretamente avrebbero desiderato convertirci alla loro fede. Fortuna che Atariani si nasce e non lo si diventa (anche se debbo ritenere di essere in errore perché da quando l'Atari è passato al sig. Tramiel i nostri computer ce li invidiano un po' tutti) per cui i bocconi

amari che i nostri antagonisti ci obbligavano a mandar giù non hanno scalfito minimamente la nostra fede nel marchio Atari. Ed oggi, con l'avvento della serie ST, dei vari 520, 1040, Mega 1, 2 e 4, dei PC IBM compatibili, della stampante laser, dell'hard disk da 20 Mega, ecc., l'Atari è entrata di prepotenza e con giusto diritto (leggi prezzo+qualità) a far parte dell'élite mondiale delle case di hardware a 16/32 bit.
Non mi dilungo oltre per non occupare spazio prezioso in questa importante "Rivista di Atari", ma prima di congedarmi (momentaneamente spero) da Voi, ho da formulare una richiesta che spero possiate esaudire: ho da qualche mese acquistato una stampante Atari 1029, e tranne qualche breve listato di programma Basic non sono riuscito ad ottenere nulla. Ho visto da un amico Atariano che esiste un programma, e precisamente il "Print-Shop" della Broderbund, che fa cose strabilianti, ma solo sulla sua stampante (possiede una Mannesmann). Sulla mia Atari 1029 non succede proprio nulla. Ora mi domando: possibile che l'Atari non abbia pensato ad un rifacimento di tale programma che si adatti alle sue stampanti? Se siete in grado di venirmi incontro, o se potete lanciare un appello attraverso la Vostra Rivista, ve ne sarei enormemente grato. Sarei anche contento se mi poteste indicare un programma alternativo a quello indicato, ma che funzioni bene sulla mia stampante.
Nel caso vi fosse in commercio quanto RichiestoVi, Vi sarei grato se mi indicaste il modo di reperirlo.
Al grido di "Forza Atari!!!" Vi saluto e resto in ansiosa attesa di una risposta circa il mio quesito. Guido Uliano
Via Nolana, 15
80045 - Pompei (NA)

Finalmente abbiamo l'occasione per dire una volta per tutte qual è il nostro vero e unico nome! Noi siamo Atariani, nome derivante dall'inglese Atarians, non Ataristi. Va bene?
La risposta alle tue domande è purtroppo negativa su tutti i fronti: la tua stampante non è assolutamente compatibile con "The Print Shop" e altri programmi del genere.
Per lo meno non siamo a conoscenza di software di questo tipo utilizzabile con l'Atari 1029.
La sola cosa che puoi fare è procurarti una stampante Epson o compatibile con relativa interfaccia. In ogni modo, se qualche lettore potrà esserti d'aiuto speriamo che si metta in contatto con te.

Spettabile Redazione,
sono molto contento che finalmente sia uscita una rivista del tutto dedicata all'Atari.
Io sono in possesso di un 800 XL e da molto tempo (circa due anni) sono alla ricerca del ComputerEyes (interfaccia digitale per le immagini da telecamera) e del Voice Master (sintetizzatore vocale) e sono riuscito a trovare soltanto il S.A.M. (programma che permette di far parlare il mio amato 800 XL).
Vorrei quindi sapere se è possibile trovare l'interfaccia digitale ed il sintetizzatore vocale e se vi è possibile dirmi il prezzo e come averli (magari anche richiedendoli all'estero).
Inoltre, per aiutare gli amici di Atari, vi mando alcune Poke da me utilizzate:
POKE 731,255 - disattiva il "Click" della tastiera
POKE 731,0 - riattiva il "Click" della tastiera
POKE 842,13 - auto-RETURN
POKE 824,12 - disattiva l'auto-RETURN
POKE 622,255 - scrolling fine, modo grafico 0 (rallenta lo scrolling del video)
POKE 580,1 - attiva il tasto RESET
Per chi vuole disattivare il tasto BREAK ecco una semplice routine (ricordate che si disattiva automaticamente al cambio di grafica):

```
10 IF PEEK (16) > = 128
THEN PE = PEEK(16) - 128
20 POKE 16,PE:POKE 53774,PE
```

Lieto dell'uscita della nuova rivista, chiudo salutandovi e dicendo che chiunque voglia entrare in contatto con me può farlo scrivendo a Roberto Rossi, Via Fraz. Ville 104, Terranuova B.ni (AR) o telefonando allo 055/93254 (anche se sarà difficile trovarmi in casa).

Purtroppo le notizie riguardo il digitalizzatore non sono buone: ComputerEyes è reperibile solamente negli Stati Uniti, dove viene prodotto dalla Digital Vision (14 Oak St. - Suite 2 Needham. MA 02192 USA) alla quale potrai rivolgerti per averlo. Il prezzo, orientativo, dovrebbe aggirarsi tra le trecento e le quattrocentomila lire.
Per quanto riguarda il Voice Master ti consigliamo di rivolgerti all'Atari Club Milano, Via Spalto Maddalena 20/C c/o Centro Libra - 21052 Monza (MI); Tel. 02/5457256 (Emanuele), 02/8395964 (Daniele) e 039/835052 (Doriano).
Circola voce che da quelle parti ne esista uno, e quindi forse loro potranno darti qualche utile indicazione.

ERRATA CORRIGE

A causa di alcuni refusi tipografici, il listato del programma per la lettura e il dump di un settore logico, di Roberto Cazzaro, pubblicato a pagina 57 sullo scorso numero è risultato errato. Per darvi modo di effettuare le correzioni lo ripubblichiamo corretto e modificato in modo che il programma possa girare anche con il GSA Basic versione 1.

```
' --------------------------------------+
'  Lettura e dump di un settore logico  |
'                                       |
'      Prima versione:     5/2/1987     |
'      Revisionato il:    14/2/1987     |
'      Versione 1.Ø  :    15/4/1987     |
'                                       |
' Autore:  Roberto "Headache" Cazzaro   |
'                                       |
'  Note  : questo programma e` stato    |
'  estratto da un programma piu` esteso,|
'  completo di menu` a discesa e varie  |
'  funzioni di lettura.                 |
'                                       |
' --------------------------------------+
'
```

```
' Funzione per la lettura di un settore
' logico (da Ø a 72Ø)
' Non viene fatto nessun controllo se il
' parametro passato e` compreso nei
' limiti quindi si ponga attenzione a
' non inserire Logical maggiore di 73Ø!

Deffn Read_sect(Logical)=Xbios(8,Buf_high
,Buf_low,Ø,Ø,Devno,Logical Mod 9+1,Logic
al Div 9,Sideno,1)
'
' Dimensionamento Buffer e
' inizializzazione variabili
'
Dim A%(256)
Buffer=Varptr(A%(Ø))
Buf_high=Buffer Div 65536
Buf_low=Buffer Mod 65536
Devno=Ø
Sideno=Ø
Er_flag=False
Ex_flag=False
Logical=Ø
'
True=-1
False=Ø
'
'            MAIN
'
Cls
Repeat
  Ret=Fn Read_sect(Logical)
If Ret=Ø Then
  Begin:
  Ex_flag=False
  Gosub Dump(Logical)
  '
  '   Ex_flag indica la condizione di
  ' uscita anomala dalla procedura DUMP
  '
  If Ex_flag=False
    Gosub Wait
  Endif
  '
  '   qui inizia il controllo del tasto
  ' premuto (manca l'istruzione CASE !)
  '
  If R=2ØØ Or R=2Ø8 Then
    Goto Begin
  Endif
  If R=2Ø3 Then
    If Logical>Ø Then
      Dec Logical
    Endif
    Goto Again
  Endif
  If R=2Ø5 Then
    If Logical<719 Then
      Inc Logical
    Endif
    Goto Again
  Endif
  If R=199 Then
    Logical=Ø
    Goto Again
  Endif
  If R=27 Then
    Er_flag=True
    Goto Again
  Endif
 '
' Qui inizia il trattamento del numero
' (necessariamente a 3 cifre !)
'
  Print At(2,2);R-48;"  "
  Temp=1ØØ*(R-48)
  Gosub Wait
  If R<48 Or R>57 Then
    Goto Begin
  Endif
     Print At(3,2);R-48
     Temp=Temp+1Ø*(R-48)
     Gosub Wait
     If R<48 Or R>57 Then
       Goto Begin
     Endif
     Print At(4,2);R-48
     Temp=Temp+(R-48)
     If Temp>719 Then
       Goto Begin
     Endif
     Logical=Temp
  Else
     Er_flag=True
     Mesg$="Errore numero "+Str$(Ret)
+".| |Ritorno all'editor"
     Alert 3,Mesg$,1," EDITOR ",Dummy
  Endif
  Again:
Until Er_flag=True
Edit
'
'
'
Procedure Dump(Logic_sect)
  Gosub Intest
  Counter=2
  I=Buffer
  '
  '   Qui bisognerebbe usare un ciclo
  '   For-Next ma il GFA Basic non
  '   permette di saltare dentro e fuori
  '   i cicli, cosi`...
  '
  Cicle:
  Print Using "####",I-Buffer;
  Print " - ";
  '
  '   Stampa i codici esadecimali
  '
  For J=Ø To 15
    Code=Peek(J+I)
    A$=Right$("Ø"+Hex$(Code),2)
    Print A$;" ";
  Next J
  Print " - ";
  '
  '   Stampa i caratteri ASCII standard
  '   compresi fra 32 e 127
  '   oppure un punto
  '
  For J=Ø To 15
    Code=Peek(J+I)
    If Code<32 Or Code>127 Then
      Print ".";
    Else
      Print Chr$(Code);
    Endif
  Next J
  '
  '   Controlla che non ci sia scrolling
  '   dei dati
  '
  Inc Counter
  If Counter<24 Then
    Print
  Else
    I=Buffer+144
    Counter=2
    Gosub Wait
    '
    '   aspetta la pressione di un tasto
    '   e ritorna il controllo a main
    '   se il tasto premuto e` diverso da
    '   freccia in basso
    '
    If R<>2Ø8 Then
      Ex_flag=True
      Goto Ex_loop
    Endif
    Gosub Intest
  Endif
  If I-Buffer<496 Then
    Add I,16
    Goto Cicle
  Endif
  Ex_loop:
Return
'
' La seguente procedura aspetta la
' pressione di un tasto e ne ritorna il
' valore nella variabile R; e` piu`
' comoda dell'INPUT e permette di
' riconoscere anche i tasti speciali
' (frecce, tasti funzione ecc.).
' Accetta solo i seguenti caratteri :
'
' (fra parentesi viene dato il codice)
'
' freccia in alto     (2ØØ)
' freccia in basso    (2Ø8)
' freccia a destra    (2Ø5)
' freccia a sinistra  (2Ø3)
' CLR HOME            (199)
' Esc                 (27)
' oppure un numero  (codici da 48 a 57)
'
Procedure Wait
  Repeat
    R=Gemdos(7)
    Rh=(R Div 65536) Mod 256
    R=R Mod 256
    If Rh>=71 And Rh<=9Ø Then
      R=Rh+128
    Endif
  Until R=27 Or R=2ØØ Or R=2Ø8 Or R=2Ø3
 Or R=2Ø5 Or R=199 Or (R>=48 And R<=57)
Return
'
'  Disegno intestazione
'
Procedure Intest
  Cls
  Print "sect |                       dati esade
cimali                   |     caratteri"
  Print Using "####",Logic_sect;
  Print " | ";
  For K=Ø To 15
    Print Right$("Ø"+Hex$(K),2);" ";
  Next K
  Print " |"
  Print "-------------------------------
---------------------------------------
-"
Return
```

ATARI NEWS

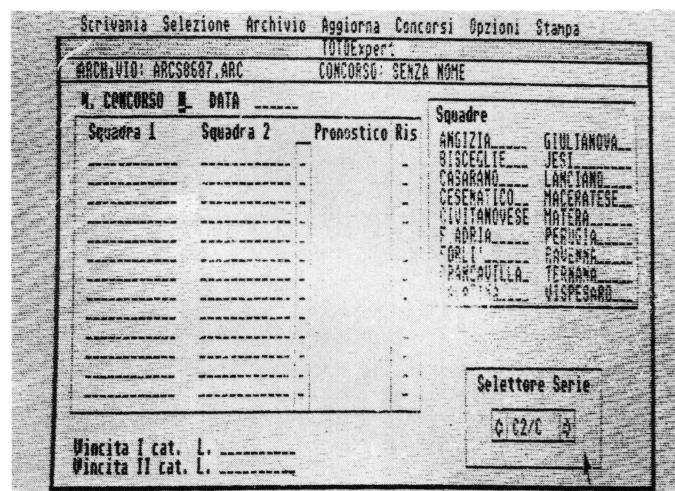
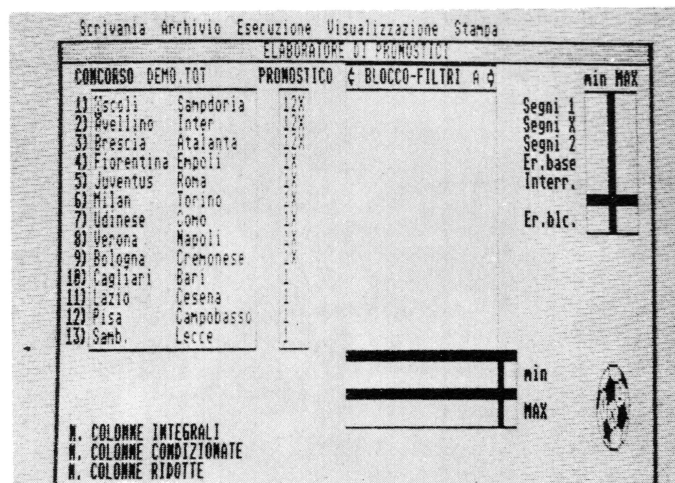
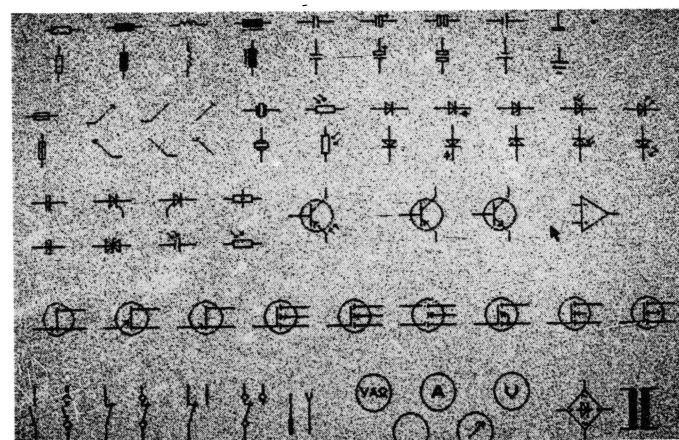
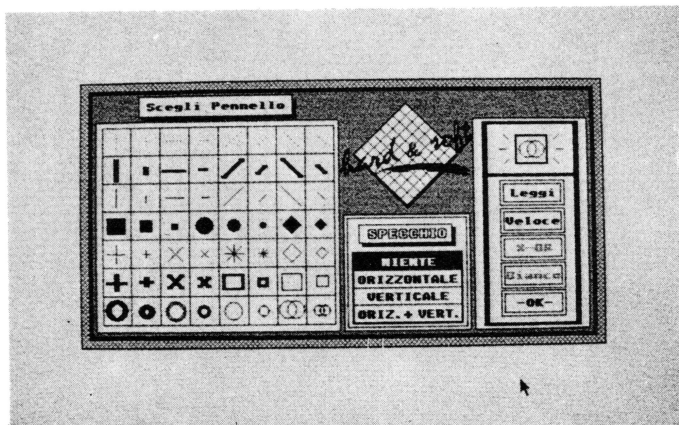
▶ Expert e Cat/Paint le maggiori novità dalla Hard & Soft

Grosse novità per il mondo Atari dalla Hard & Soft di Terni. La ditta umbra ha recentemente immesso sul mercato nuovi prodotti hardware e software per la serie ST sviluppati esclusivamente "in casa".

Le due proposte software sono costituite da un programma CAD, abbinabile anche ad una tavoletta grafica, e da un programma di sviluppo pronostici adattabile al Totocalcio, al Totip e all'Enalotto.

Il programma grafico si chiama CAT/Paint e costituisce un sensibile miglioramento anche rispetto al già eccellente Degas Elite grazie ai molti effetti speciali ottenibili.

CAT/Paint viene fornito dalla Hard & Soft unitamente ad una vasta libreria di simboli utilizzabili nei disegni.

L'unico neo di questo programma è rappresentato dal fatto che per il momento CAT/Paint è disponibile esclusivamente in versione ad alta risoluzione e quindi monocromatica. Il problema, ci hanno però garantito, verrà sicuramente risolto nella prossima versione che sarà perciò a colori. CAT/Paint è venduto a richiesta abbinato ad una tavoletta grafica, sempre firmata Hard & Soft, che può essere poi sfruttata anche per altri programmi. I prezzi di questi due prodotti saranno allineati alla strategia di vendita dei computer Atari e dovrebbero quindi avere un prezzo base piuttosto contenuto.

Il programma per lo sviluppo dei pronostici si annuncia molto sofisticato e adatto anche alle ricevitorie. Di questo programma oltre a ciò non si sa ancora molto, ma avremo occasione in futuro di parlarne ancora.

Sempre da Hard & Soft sono in arrivo due digitalizzatori: RealTizer ed Expert.

Il primo è in grado di visualizzare immagini immobili in tempo reale con una risoluzione di 320 punti per 200, il secondo invece impiega alcuni secondi, ma la risoluzione è settabile dall'utente da un minimo di 256×256 pixel fino ad un massimo di 1024×512! Inoltre Expert consente di operare con ben 128 tonalità diverse di grigio. Una risoluzione così alta è l'ideale per lavori sofisticati e serve ad ottimizzare l'immagine video. Altra particolarità è la possibilità di controllare direttamente da programma la luminosità e il contrasto di ogni singola immagine.

Se a tutto ciò aggiungiamo che le immagini ottenute con i due apparecchi sono totalmente compatibili con TUTTI i programmi grafici, compreso l'ultimo arrivato CAT/Paint, si può facilmente capire quale sia il livello di

Totoexpert è il nuovo programma della Hard & Soft per i totogiocatori. Si annuncia molto sofisticato e adatto quindi anche alle ricevitorie.

Due schermate di Cat/Paint. Interessante è la libreria di simboli che viene fornita nel dischetto insieme al programma.

questi digitalizzatori.
Per ulteriori informazioni: Hard & Soft - Via Bolzello, 2 - 05100 Terni - Tel. 0744/46658.

► Sette giochi per tutti i gusti

La Mastertronic continua ad immettere sul mercato novità software per le macchine Atari, sia per 8 bit, sia per ST. Il settore dei videogiochi sembra essere uno dei più attivi negli ultimi tempi, e la casa varesina ha aggiornato il suo catalogo con sette nuovi programmi di questo tipo, rimanendo fedele alla politica dei prezzi contenuti.
Per i computer della serie 800 e 130 sono ora disponibili Countlet, Arkanoid, Mutant Camels e Power Bovver, questi ultimi due posti in vendita con Nice Price, ovvero a 5000 lire cad.
Tre nuovi arrivi per le macchine ST completano il quadro delle novità in casa Mastertronic: Mean 18, Micro Cross e Arkanoid.
Il primo è una nuova versione del golf che certamente non mancherà di interessare gli appassionati del genere, Micro Cross è un divertente gioco d'azione che si svolge su di una scacchiera e che consiste in una gara su pattini a rotelle resa difficile da ogni sorta di ostacolo posto sul percorso e Arkanoid è un rifacimento realizzato veramente molto bene del noto gioco che consisteva nel distruggere un muro di mattonelle con una pallina e una racchetta.
Descritto a parole può sembrare banale, ma in realtà il game si rivela divertente e coinvolgente.
Per ulteriori informazioni: Mastertronic - Via Mazzini, 15 - 21020 Casciago (Varese) - Tel. 0332/212255.

► Lago per l'office automation

Alcune primizie interessanti ci vengono proposte dalla Lago, che è sempre una delle più attive importatrici di software per Atari. Atteso da tempo, arriva il tedesco GSA Draft, che però nel nostro paese verrà commercializzato nella versione inglese curata dalla GlenTop.
Questo programma grafico si presenta molto bene con una serie di feature che lo rendono appetibile ad una vasta schiera di potenziali utenti e dovrebbe essere immesso sul mercato ad un prezzo contenuto al di sotto delle duecentomila lire.
Sempre nella stessa categoria di prezzo si collocherà Trim Base, un database relazionale di cui però al momento non sappiamo molto.
Vedremo...
Una serie di programmi destinati all'office automation sono la carta che Lago gioca nel settore del software integrabile: Word Writer, Data Manager e Swift Calc sono tre programmi completamente compatibili tra loro che dovrebbero risolvere qualsiasi problema di trasferimento dati ad un costo molto contenuto.
Il word processor, il database e lo spreadsheet prodotti dalla Timeworks dovrebbero costare ciascuno 99.000 lire.
Ancora dalla Lago tre giochi: l'ormai famoso Barbarians, con una grafica splendida, l'altrettanto celebre Star Trek e Turbo GT, simulatore di corsa.
Per ulteriori informazioni: Lago - Viale Masia, 79 - Como - Tel. 031/300174.

COMPUTEREYES

ESISTONO PRODOTTI CHE SFRUTTANO A FONDO LE CAPACITÀ GRAFICHE DEGLI ATARI 8 BIT; PER ESEMPIO IL DIGITALIZZATORE

DI EMANUELE BERGAMINI

Abbiamo già visto un digitalizzatore per ST, ora tocca agli 8 bit. COMPUTEREYES appare come una piccola scatola dalla quale spuntano due spinotti che vanno inseriti nelle porte 1 e 2 dei joystick, due manopole per regolare l'immagine e una presa a cui si collega la nostra sorgente video, sia essa una telecamera o un videoregistratore.
Nelle confezione troviamo un manualetto sufficientemente esplicativo (in inglese) e un disco contenente il programma di controllo, che è scritto (in parte) in basic.
Carichiamo il disco. Dopo alcuni minuti comparirà il menu principale, con varie opzioni. Esaminiamole una ad una: – ADJUST SYNC – serve a sincronizzare l'apparecchio con la sorgente video, girando la manopola di sinistra.
BRIGHTNESS – serve a regolare il fuoco e la luminosità del segnale, girando la manopola di destra. Queste due operazioni devono essere fatte all'inizio, poi non più. Ora possiamo finalmente catturare le immagini.
L'opzione NORMAL CAPTURE è la più facile da usare. In soli 6 secondi apparirà sul video l'immagine. Questa opzione effettua una scansione dell'immagine esaminando la luminosità di ogni punto che viene confrontata con un valore campione: ogni punto più chiaro apparirà bianco, mentre quelli più scuri saranno neri.
4 LEVEL CAPTURE – lavora in maniera simile alla precedente, ma effettua 4 scansioni differenti su 4 diverse soglie di grigio. L'immagine finale sarà composta dalla somma delle 4 scansioni. Analogamente, l'opzione 8 LEVEL CAPTURE effettua 8 scansioni diverse, producendo non più immagini con forte contrasto ma con bellissimi effetti di sfumature.
Naturalmente ci vuole più tempo per catturare un'immagine, cioè rispettivamente 25 e 50 secondi.
Queste tre opzioni producono immagini che utilizzano l'alta risoluzione della grafica 8 monocromatica, quindi le immagini sono compatibili con tutti i programmi grafici che utilizzano la grafica 8. Ma le potenzialità grafiche dell'Atari non si fermano qui, ed è possibile digitalizzare un'immagine in grafica 15 (meglio conosciuta come grafica 7 1/2). Le opzioni CONTRAST CAPTURE producono, in 18 secondi, immagini che usano i quattro colori disponibili come sfumature di grigio con contrasti più o meno forti. Le immagini così realizzate sono perfettamente compatibili con i maggiori pacchetti grafici, compreso l'AtariArtist con il quale possiamo colorare le immagini e, con una stampante grafica, persino portarle su carta. È addirittura possibile stamparle a colori.
Le altre opzioni del menu permettono di stivare o recuperare un'immagine su disco (occupare 62 settori). L'opzione VIEW mostra l'ultima immagine selezionata (digitalizzata o caricata da disco), mentre CATALOG mostra la directory. Molto utile è l'opzione HELP che fornisce un minimo di indicazioni su ogni opzione. Per attivarla si deve premere "?" seguito dal tasto dell'opzione di cui vogliamo la spiegazione, premendo semplicemente return torniamo al menu.
Cosa si può dire per concludere? Questo apparecchio contribuisce a rendere veramente completo il parco accessori del nostro 8 bit, alla faccia di quelli che credono che serva solo per giocare a Pac-Man. Purtroppo chi vuole aggiudicarsi COMPUTEREYES si trova costretto a comprarlo direttamente dall'America, spendendo qualcosa come tre/quattrocentomila lire, ma pensate al divertimento di poter digitalizzare la propria ragazza o il gatto di casa che dorme. Chi ama la grafica non può proprio farne a meno.
Prodotto da:
DIGITAL VISION, 14 Oak St. – Suite 2 Needham.Ma 02192 USA

Un'immagine realizzata con Computereyes e successivamente stampata con Okimate 20

TIPS & TRICKS

DI MAURO PAVONE

I SET DI CARATTERI ATARI

Se possedete un Atari Xl e XE, nella vostra ROM, a partire dalla locazione 52224, per una lunghezza di 1K, è riservato nella memoria il cosiddetto "set internazionale" di caratteri, che dispone di un vasto campionario di lettere corrispondenti alle esigenze europee (ad esempio le vocali accentate, come nelle macchine per scrivere). Per attivarlo, digitate "POKE 756,204", per tornare al set normale (quello con i caratteri grafici), "POKE 756,224".
Quando il set internazionale è in uso, otterrete i "nuovi" caratteri premendo, con il CONTROL, un tasto alfabetico.

SULLA TASTIERA

Con la pressione di un tasto sapete che il vostro Atari produce un "click". Se ciò vi dà fastidio, con un POKE 731,255 lo eliminerete. Per riattivarlo usate POKE 731,0.
Quando invece tenete abbassato per un certo periodo di tempo il tasto, il carattere in questione verrà ripetuto sullo schermo finché non lascerete il tasto. Per variare il periodo, inserite il valore desiderato nella locazione 729, che inizialmente è 48 (sessantesimi di secondo). Un valore alto indica un tempo lungo, uno basso un tempo breve, con lo 0 si disinserisce la ripetizione automatica.
La locazione 730, controlla il numero di caratteri visualizzabili in un secondo durante la ripetizione automatica. Inizialmente contiene un 6. Con uno 0, anche in questo caso, si elimina momentaneamente la ripetizione.
La tavola di definizione della tastiera è l'elenco dei caratteri appartenenti ad ogni tasto. È possibile ridefinirla, intervenendo sull'ordine iniziale inserendo i 192 dati necessari in una zona libera di memoria e sistemando di conseguenza i puntatori alle locazioni 121 e 122 che normalmente puntano la locazione 64337.
I primi 64 dati riguardano i tasti premuti senza SHIFT, i secondi i tasti premuti con SHIFT ed infine, l'ultimo gruppo, quelli premuti con CONTROL, secondo il seguente ordine:

l	j	;	F1	F2	k	+	*
o	...	p	u	RETURN	i	–	=
v	...	c	F3	F4	b	x	z
4	...	3	6	ESC	5	2	1
,	SPACE	.	n	...	m	/	inverse
r	...	e	y	TAB	t	w	q
9	...	0	7	BACKS	8	<	>
f	h	d	...	CAPS	g	s	a

CONVERSIONE DEL BASIC

I basic installati nei vostri ATARI possono essere diversi gli uni dagli altri; nei vecchi 400 e 800 c'è il cosiddetto Basic A, che è la prima versione. La maggior parte degli Atari 800 e 600 XL, invece, è stata dotata del nuovo Basic B che non è riuscito a supplire la carenza della prima versione che, nonostante fosse un ottimo prodotto, presentava un fastidioso "bug" denominato familiarmente "keyboard lockup". Anche il Basic B presenta un "bug" che porta a malfunzionamenti quali l'incorrere nell'errore 9 o il blocco del sistema. Tutto questo è sparito nella versione C che è stata adottata, negli XL, a partire dal Marzo del 1985. Il programma che segue ha lo scopo di convertire il vostro basic da B in C. Per sapere che Basic avete, controllate la locazione 43234 e verificate la versione che sarà A, B o C a seconda del valore che può essere 162,96 o 234.

```
10 FOR I=0 TO 43:READ A:POKE 16384+I,A:NEXT I:A=USR(16384)
20 DATA 104,169,0,133,203,169,160,133,204,162,32,160,0,177,203,72
30 DATA 169,255,141,1,211,104,145,203,169,253,141,1,211,136,208,237
40 DATA 230,204,202,48,6,208,230,160,0,208,226,96
50 FOR J=1 TO 13:READ A,B:POKE A,B:NEXT J
60 DATA 54017,255,43231,234,43232,240,43233,17
70 DATA 43234,234,47913,0,49139,0,49140,0,49141,0,49142,0,49143,0,49144,0,49145,0
```

A SCUOLA DI BASIC

CONCATENAMENTO DI PROGRAMMI

DI EMANUELE BERGAMINI

COME SCRIVERE PROGRAMMI CONCATENATI CON IL BASIC ATARI

Il Basic Atari permette di concatenare i programmi usando il comando RUN "dev" ("dev" indica la periferica, ovvero "C:" per il registratore, "D: nomefile.ext" per il driver), a condizione che il programma da caricare sia stato salvato precedentemente con SAVE"dev". Tuttavia il Basic Atari riserva sempre molte sorprese: infatti ci sono altri metodi validi per concatenare dei programmi. Per poter caricare un programma dobbiamo prima conoscere come viene salvato. Ci sono tre modi per salvare un programma:
1) usando il comando SAVE"dev". In questo modo i programmi vengono salvati in forma tokenizzata (cioè codificati, dove ad ogni istruzione corrisponde un codice). Questo è il modo comunemente usato per salvare i programmi su disco. Il sistema dei "tokén" (in inglese "gettoni") permette di risparmiare molto spazio su disco, perché i codici di ogni comando o istruzione sono composti da non più di un paio di bytes. L'istruzione APPEND, per esempio, invece di occupare 6 bytes ne occupa solo 2. Il salvataggio su cassette risulta esser abbastanza lento, poiché passa un certo tempo tra la trasmisione di un blocco di dati e un altro.
2) Usando il comando CSAVE. Questo comando salva il programma solo sul registratore in forma tokenizzata, e si differenzia da SAVE"C:" solo per il fatto che è più veloce poiché i blocchi di dati non sono interposti da spazio. Per ricaricare un programma si usa CLOAD.
3) Usando il comando LIST"dev". In questo caso la periferica di uscita può essere anche la stampante o lo schermo. Il programma, che è mantenuto in memoria in forma tokenizzata, viene convertito in formato ATASCII leggibile e mandato alla periferica di output, esattamente come apparirebbe sul video. Questo è il modo più lento per caricare o salvare un programma.
Sappiamo già che concatenare programmi salvati con SAVE non crea problemi: basta usare RUN"dev" e il programma viene caricato ed eseguito.
Il caricamento è però distruttivo perché il programma chiamante viene cancellato dalla memoria come se avessimo dato un NEW: anche le variabili vengono quindi cancellate e non possono essere utilizzate nel programma chiamato.
Per chi ha solo il registratore può essere noioso usare il "lento" SAVE quando si

potrebbe usare un metodo più rapido per salvare i propri programmi.
I più smaliziati che possiedono un drive possono anche usare il dischetto come memoria dove mantenere tutte le variabili che vogliamo salvare: il primo programma apre un file in scrittura dove scrive le variabili, poi fa partire il secondo programma con RUN"D:..." (che è più veloce di ENTER), il quale aprirà in lettura il file da dove recupererà le variabili che gli servono.
Ma come si fa a concatenare un programma salvato con CSAVE? Non si può utilizzare CLOAD all'interno di un programma, perché fa terminare qualsiasi esecuzione alla fine del caricamento (così come LOAD e ENTER): saremmo dunque costretti a dare un RUN manualmente. In Atari Basic non esiste un'istruzione per caricare e far partire un programma salvato con CSAVE; il listato 1 presentato in questa pagina è una routine in linguaggio macchina che provvede alla mancanza di un comando "CRUN".
Il comando ENTER"dev" carica un programma salvato con LIST, ma a differenza di LOAD e CLOAD non è distruttivo. Questo comando agisce infatti prendendo dalla periferica le linee di comandi come se fossero digitati da tastiera. Il vecchio programma rimane in memoria, ma se alcune linee di istruzioni hanno il numero di linea uguale a quello delle linee già presenti nella memoria stessa, queste verranno sostituite da quelle nuove.
Se il file caricato contiene dei comandi senza numero di linea (creabili con un qualsiasi editor di linea o un WP, o aggiunti con dei PRINT# nel file aperto in modo APPEND), questi verranno eseguiti come domandi diretti. Anche le variabili verranno mantenute perché il secondo programma si è fuso con il primo e non si è sostituito ad esso.
Ma come si fa a continuare l'esecuzione del programma dopo aver caricato la seconda parte visto che ENTER fa terminare l'esecuzione?
I casi sono come al solito due: o il programma da concatenare termina con un comando diretto come un RUN o un GOTO a una certa linea, oppure deve utilizzare una particolarità della locazione 842 che normalmente contiene il valore 12; se questo valore viene portato a 13, il computer entra nel "MODO RETURN" e si comporta come se il tasto RETURN fosse sempre premuto: ogni scritta sullo schermo può quindi essere utilizzata come comando. Questa locazione infatti contiene il modo di utilizzo del canale IOCB 0, cioè quello utilizzato dal Basic come editor, ed è normalmente posto a 12 (modo input/output). Portandolo a 13 si porta il bit 0 a 1, attivando il modo di lettura concorrente usato con le porte RS-232, ma usato con l'editor produce l'effetto che sappiamo. Questo stesso principio è stato usato nella realizzazione del ControList, che utilizza un canale aperto sull'editor in modo 5, cioè lettura (mod 4) a cui viene aggiunto il valore 1 per settare il bit 0.
Il listato 2 utilizza questa caratteristica per fare un concatenamento automatico con ENTER. Quest'ultimo può essere usato nel caso in cui un programma principale debba richiamare parecchie subroutine: basta salvare una alla volta tutte le subroutine con gli stessi numeri di linea, così quando una viene caricata si sostituisce all'altra già presente, e può utilizzare le stesse variabili senza problemi. In ogni caso la scelta del metodo da utilizzare dipende soprattutto dal tipo di programma e dalla periferica a disposizione: chi utilizza il registratore avrà anche un problema in più, ossia quello del posizionamento del nastro. La particolarità della locazione 842 può inoltre essere sfruttata per effettuare altri comandi senza farli digitare direttamente da chi usa il programma stesso.

LISTATO 1

```
100 DIM CRUN$ (40)
110 FOR T=1 TO 19
120 READ C
130 CRUN$(T,T)=CHR(C)
140 NEXT T
150 POKE 764,32: REM SIMULA
    LA PRESSIONE DI UN TASTO
160 X=USR(ADR(CRUN$))
170 DATA 162,253,154,169,183,
    72,169,84,72,169,4,32,182,
    187,169,255,76,4,187
```

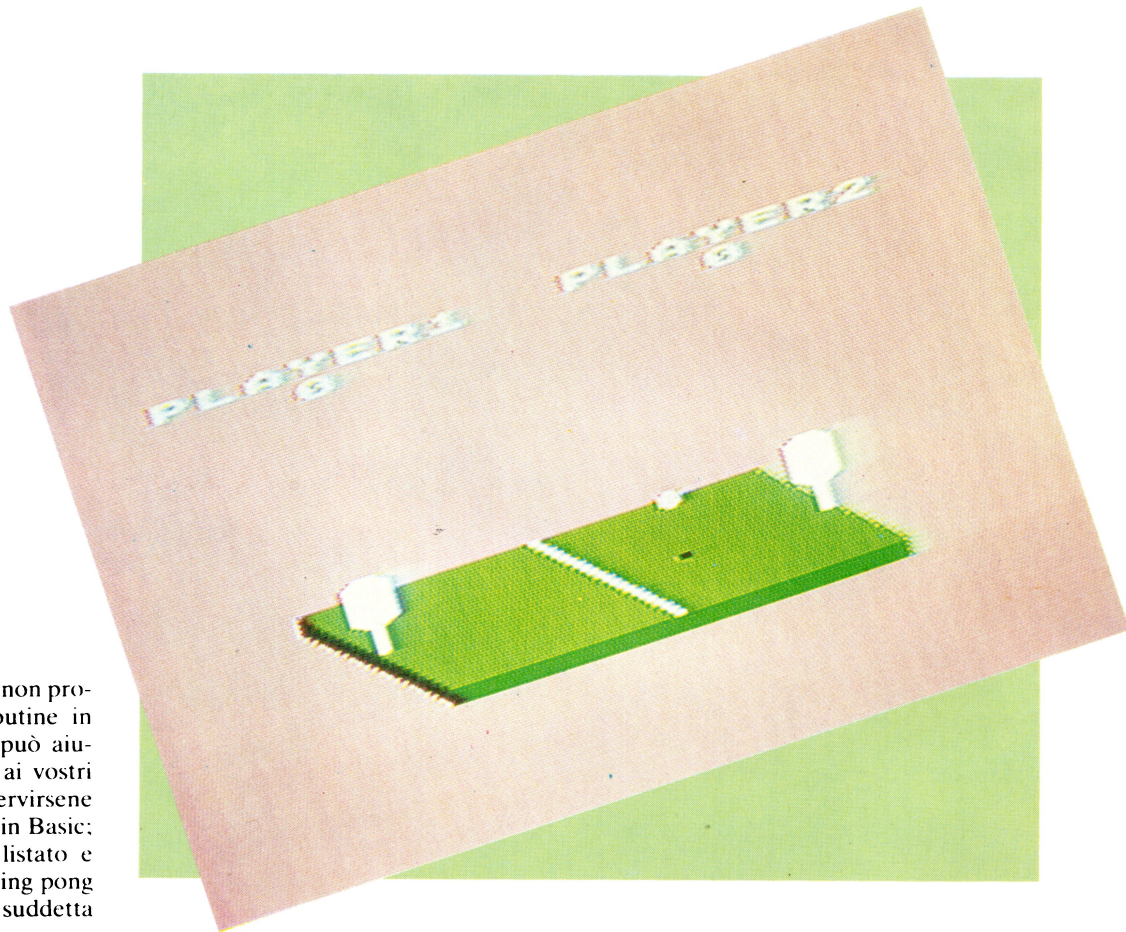
LISTATO 2

```
100 GRAPHICS 0
105 PRINT CHR$(125)
110 POSITION 2,3
120 PRINT "ENTER";
    CHR$(34);"D:PROGRAM2"
130 PRINT "CONT"
150 POSITION 2,0
160 POKE 842,13:STOP
170 POKE 842,12
180 REM IL PROGRAMMA
    CONTINUA DA QUI
```

XPONG

SPAZIO BASIC

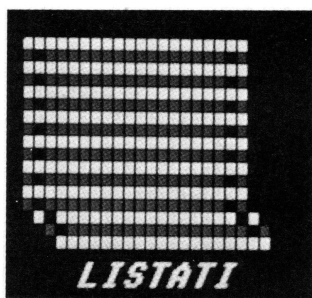
DI MAURO PAVONE

Per i programmatori e per i non programmatori, ecco una routine in linguaggio macchina che può aiutarvi a dare un tocco di classe ai vostri programmi. I primi potranno servirsene per inserirla nei loro capolavori in Basic; i secondi potranno digitare il listato e giocare con XPONG, un mini-ping pong realizzato con l'ausilio della suddetta routine.
Questa serve alla gestione degli spostamenti orizzontali e soprattutto verticali dei quattro Player messi a disposizione dall'hardware dell'Atari. Sapendo che il movimento orizzontale si realizza semplicemente mettendo la coordinata del player 0-3 nelle locazioni 53248 - 53251 e che il movimento verticale si ottiene con lo spostamento del blocco dati, relativo alla figura, nell'area dedicata, si può intuire il problema di lentezza del secondo tipo di movimento che ho citato.
La routine ha proprio il compito di porre rimedio, con la massima semplicità, a questo inconveniente, con l'ausilio di un linguaggio veloce.
Per utilizzarla nei vostri programmi, basta settare la RAM TOP alla locazione

```
0 REM ******************************
1 REM *                            *
2 REM *    XPONG (C) 1987 by MaP   *
3 REM *                            *
4 REM ******************************
5 GOSUC 9000
10 REM .........................VRBL
20 X1=10:X2=10:XP=11:YP=1:ZP=2:XK=0:YK=1:ZK=3:P1=0:P2=0:HI=0:PT=15
50 REM .........................GAME
60 S0=STICK(0):S1=STICK(1)
70 X1=X1+(S0=7)+(S0=5)+(X1<0)-(S0=11)-(S0=10)-(X1>20)
90 K=USR(1536,0,RR,PM,X1+76,120+X1*2)
100 X2=X2+(S1=7)+(S1=5)+(X2<0)-(S1=11)-(S1=10)-(X2>20)
120 K=USR(1536,3,RR,PM+768,X2+154,120+X2*2)
130 XP=XP+XK:YP=YP+YK:ZP=ZP+ZK
140 IF XP<1 OR XP>19 THEN XK=-XK
150 IF YP>20 OR YP<1 THEN YK=-YK:IF YP=0 THEN GOSUB 9500
170 K=USR(1536,1,PP,PM+256,XP+75+ZP,140+XP*2-YP):K=USR(1536,2,OO,PM+512,XP+75+ZP
,140+XP*2)
180 IF ZP<2 OR ZP>75 THEN GOSUB 500
490 GOTO 50
500 REM .........................CLL.
510 IF ZK<0 AND ABS(XP-X1)<3 THEN 540
520 IF ZK>0 AND ABS(XP-X2)<3 THEN 540
```

```
530 GOTO 600
540 GOSUB 9500
550 IF XP>10 THEN XK=-RND(0)*2
560 IF XP<10 THEN XK=RND(0)*2
570 ZK=-ZK
580 RETURN
600 REM .........................OUT.
610 XP=XP+XK*(XP>0 AND XP<20):YP=YP+ABS(YK):ZP=ZP+ZK:K=USR(1536,1,PP,PM+256,XP+7
5+ZP,140+XP*2-YP)
620 SOUND 0,ZP+80,10,INT(RND(0)*3)+2:IF ZP>-60 AND ZP<130 THEN 610
630 SOUND 0,0,0,0
640 IF ZK<0 THEN P2=P2+1
650 IF ZK>0 THEN P1=P1+1
660 POKE 87,0:POSITION 0,13:PRINT "  PLAYER1  PLAYER2        ";P1;"          ";P2:P
OKE 87,7
670 XP=11:YP=1:ZP=30:XK=0.5:YK=1
680 IF ZK<0 THEN ZK=ZK-0.1
690 IF ZK>0 THEN ZK=ZK+0.1
700 ZK=-ZK
710 IF P1>P2 THEN HI=P1
720 IF P2>P1 THEN HI=P2
730 IF HI<PT THEN GOTO 50
740 FOR T=1 TO 1000:NEXT T:GOTO 9400
9000 REM .........................SCRN
9010 RAM=PEEK(106)-16:POKE 106,RAM
9020 GRAPHICS 23:SETCOLOR 0,0,10:SETCOLOR 1,12,4:SETCOLOR 2,12,2:SETCOLOR 4,2,6:
POKE 752,1:POKE 82,0
9030 DL=PEEK(560)+PEEK(561)*256:POKE DL+18,6:POKE DL+19,6
9040 POKE 87,0:POSITION 0,13:PRINT "     XPONG BY MAP",:POKE 87,7
9050 COLOR 3:FOR K=0 TO 2:PLOT 30,51+K:DRAWTO 50,71+K:DRAWTO 130,71+K:NEXT K
9060 COLOR 2:FOR X=30 TO 110:PLOT X,50:DRAWTO X+20,70:NEXT X
9070 COLOR 1:PLOT 70,50:DRAWTO 90,70:PLOT 71,50:DRAWTO 91,70
9080 COLOR 0:PLOT 0,0
9100 REM .........................L.M.
9110 FOR A=1536 TO 1606:READ B:POKE A,B:NEXT A
9120 DATA 104,104,104,170,104,141,58,6,104,141,57,6,104,141,49,6,141,61,6,104,14
1,48,6,141,60,6
9130 DATA 104,104,157,0,208,104,104,141,71,6,162,0,160,0,169,0,204,71,6,240,9,15
3,0,0,200,192,255,208,243,96
9140 DATA 189,0,0,153,0,0,200,232,224,28,208,244,76,40,6
9200 REM .........................DATI
9210 DIM R$(28),P$(28),O$(28)
9220 FOR A=1 TO 28:READ B:R$(A,A)=CHR$(B):NEXT A
9230 DATA 60,126,255,255,255,255,255,255,255,255,255,255,255,255,255,255,126,60,
24,24,24,24,24,24,24,24,24,24
9240 FOR A=1 TO 28:READ B:P$(A,A)=CHR$(B):NEXT A
9250 DATA 24,60,60,60,60,24,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
9260 FOR A=1 TO 28:READ B:O$(A,A)=CHR$(B):NEXT A
9270 DATA 0,0,0,0,24,24,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
9300 REM .........................SETP
9310 POKE 54279,RAM:POKE 559,62:POKE 53277,3:POKE 704,14:POKE 705,10:POKE 706,0:
POKE 707,14:POKE 623,1
9320 RR=ADR(R$):PP=ADR(P$):OO=ADR(O$):PM=(RAM+4)*256
9400 REM .........................WAIT
9410 POKE 87,0:POSITION 0,13:PRINT "                              READY.    ":POKE
87,7
9420 IF STRIG(0)<>0 THEN GOTO 9420
9430 POKE 87,0:POSITION 0,13:PRINT "  PLAYER1  PLAYER2       0         0":POKE 87
,7
9440 GOTO 10
9500 REM .........................SND.
9510 FOR V=12 TO 0 STEP -2
9520 SOUND 0,80,10,V
9530 NEXT V
9540 SOUND 0,0,0,0
9550 RETURN
```

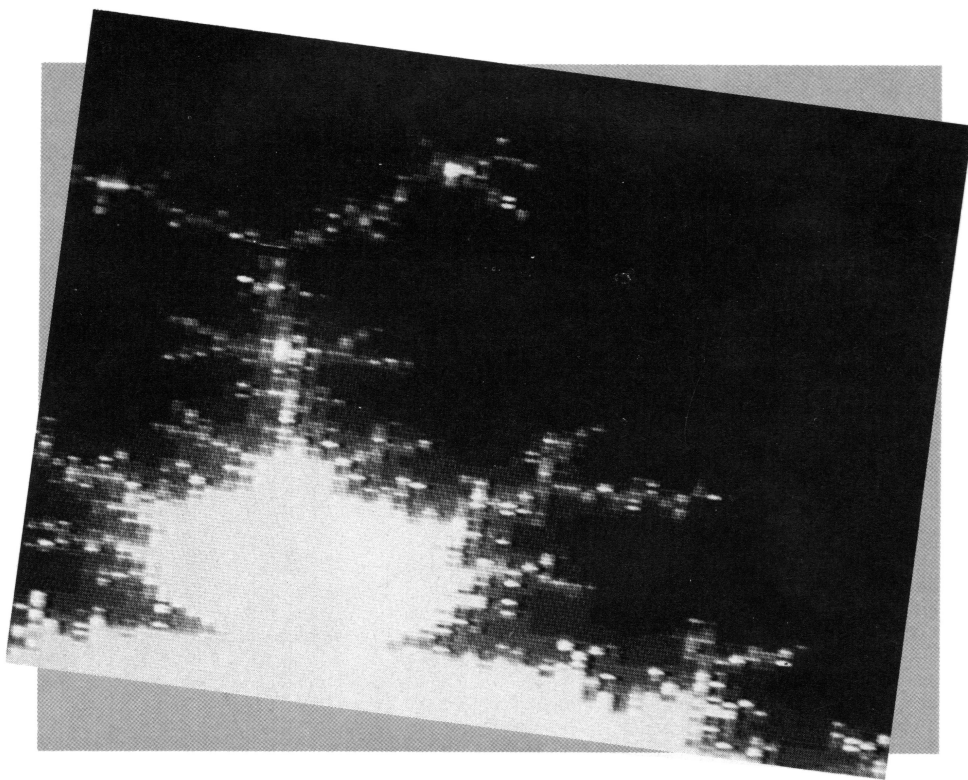
COMMENTO AL LISTATO (XPONG)

10-20	Variabili usate nel gioco
50-180	Routine principale
60-120	Controllo dei joystick e movimento delle racchette
130-170	Movimento della palla e dell'ombra, gestione del rimbalzo e della direzione
180	Controlla se la palla ha raggiunto uno dei lati del tavolo
500-580	Verifica se la racchetta ha colpito la palla, se si inverte la direzione della palla, altrimenti prosegue da 600
600-740	La palla non è stata "presa", con un ciclo viene fatta scomparire, il punteggio viene attribuito e, se non è il punteggio limite, riposiziona le coordinate della palla e ritorna al ciclo principale
9000-9080	Qui viene definito lo schermo, cambiate le due linee di display list e disegnato il tavolo
9100-9140	Vengono inseriti i dati della routine in linguaggio macchina nello locazioni 1536-1606
9200-9270	I dati delle figure vengono depositati nelle 3 stringhe: R$ per la racchetta, P$ per la palla e O$ per la sua ombra
9300-9320	Preparazione della grafica player missile
9400-9440	Stampa la scritta "ready". Quando il primo giocatore preme il tasto del suo joystick, appaiono i punteggi e il controllo passa alla routine principale
9500-9550	Generazione del suono di rimbalzo

106, in modo da riservare lo spazio sufficiente alla grafica player missile, inserire una linea che inizializzi la suddetta grafica e, infine, inserire i dati della routine in linguaggio macchina a partire dalla locazione 1536 compresa.
Per richiamare la routine si adotta un'istruzione del tipo:
X = USR (1536, numero del player, area dati della figura, area di memoria del player n. N, coordinata X, coord. Y).
La routine si occuperà automaticamente di mantenere pulita la zona di memoria di ogni singolo player, senza lasciare fastidiose "scie" nel movimento o pixel "vaganti".
Per modificare la routine secondo le vostre necessità, potete variare l'altezza della figura che, nel caso del gioco, è fissata a 28 pixel, ma che può essere diminuita o aumentata a vostro piacimento, da un minimo di un pixel a un massimo di circa 255 pixel, mettendo il valore scelto al posto del 28 (sesto dato a partire dal fondo, tra i DATA che seguono il REM... L.M.).
Il gioco che utilizza questa routine, è una semplice simulazione di ping pong, alleggerito dalle sue molteplici regole e caratteristiche, nel quale voi dovrete semplicemente non farvi sfuggire la pallina che da lenta diventerà, con l'aumentare dei punteggi, sempre più veloce.
Lo schermo raffigura un piccolo tavolo da gioco con visione tridimensionale (!!!) ed i punteggi. Le figure realizzate in grafica P-M sono rappresentate dalle due racchette, dalla pallina e dalla sua ombra. Il modo grafico scelto è il 7, senza finestra testuale, con una variazione di due linee nel display list, in modo da poter visualizzare messaggi quali il punteggio, la scritta "READY" e l'intestazione.
Per giocare bisogna essere in due, inserire i joystick personali e, per iniziare, premere il fire del joystick inserito nella porta 1.
Il punteggio limite è 15, ma può essere facilmente cambiato sostituendo alla linea 20 il valore 15 della variabile PT, con un altro da voi preferito, stando attenti ai tragici effetti che avrà sulla velocità della pallina, se fisserete, ad esempio, un punteggio massimo di 30 (velocità che si realizzerà, oltretutto, con scatti felini).
Naturalmente, vincerà chi raggiungerà per primo tale punteggio.
Questo gioco non è senz'altro il meglio che si possa realizzare nel genere, soprattutto per la velocità; un programma a livello più alto lo potete trovare tra le vostre miriade di giochi, senza però capire come è stato realizzato! Questo, invece, vuole rappresentare un modesto esempio di quello che si può ottenere con la grafica player-missile, una semplice routine e delle basi essenziali di programmazione che ognuno di voi può facilmente raggiungere.

FRACTAL MAKER

DI EMANUELE BERGAMINI

UN PROGRAMMA IN BASIC PER GENERARE FRATTALI A PIACERE

Beniot B. Mandelbrot, un ricercatore della IBM, ha recentemente sviluppato un particolare campo dello studio delle forme geometriche che ha chiamato GEOMETRIA FRATTALE. Una delle sue proprietà è quella di poter essere rappresentata graficamente con risultati sorprendenti. In questo articolo ci occuperemo di esplorare un particolare insieme di numeri complessi chiamato appunto Insieme di Mandelbrot.
Innanzitutto occorre una piccola premessa su alcun concetti matematici, primo tra tutti il concetto di numero complesso.
Un numero complesso è un numero composto da una parte reale e una parte immaginaria, indicata dal simbolo "i". 7+3i è un numero complesso, così anche −6i e 7-14i. Ogni numero complesso può essere rappresentato sul cosiddetto Piano Complesso (simile al piano cartesiano), dove l'origine è data dal numero complesso 0+0i, la parte reale individua la coordinata X, mentre la parte immaginaria individua la coordinata Y.
Per sommare due numeri complessi basta fare la somma delle due parti separatamente, mentre per moltiplicarli occorre procedere come nell'algebra ordinaria, sapendo però che i i2 è uguale a -1.
La dimensione di un numero complesso è la distanza sul piano tra esso e l'origine, ricavata applicando il Teorema di Pitagora ai lati.
Torniamo al nostro insieme. Quando applichiamo ai numeri una certa funzione ricorsiva, quelli che sono all'esterno dell'insieme scappano all'infinito, mente quelli all'interno ondeggiano. La parte più suggestiva è la zona di margine.
La funzione applicata è: Z=Z2+C, dove Z è un numero complesso arbitrario mentre C è un numero del piano. Continuando ad applicare la stessa funzione, il numero tenderà ad aumentare, fino a superare la capacità matematiche del nostro Atari. Fortuntamente i numeri che stanno nell'insieme di Mandelbrot sono tutti i numeri C che hanno dimensione finita anche dopo numerose iterazioni, così possiamo comodamente escludere gli altri.
Passiamo al nostro programma. Per poterlo usare bisogna avere un disk drive e un Atari XL o XE, infatti non funziona

correttamente sui vecchi 400 e 800 mancanti del tasto HELP.

Il menu presenta 7 opzioni. Vediamo di esaminarle una ad una:

FORMAT DISK – serve a formattare un dischetto. Non vengono fatti controlli, quindi inserite un disco vergine.

DIRECTORY – visualizza la directory completa del disco.

FINE PROGRAMMA – termina l'esecuzione del programma. Viene chiesta la conferma della decisione.

CREA FRATTALE – questa è l'opzione principale. Per creare un frattale bisogna inserire una serie di parametri iniziali: per il campo del frattale, che ha una forma quadrata, bisogna indicare l'angolo inferiore sinistro come un numero complesso (x=parte reale, y= parte imm.), seguito dalla misura del lato. Poi si deve inserire il nome del frattale (max 8 car.) che verrà poi utilizzato per salvare il frattale su disco. NON usate estensioni o nomi di periferiche.

CONTINUA FRATTALE – serve a continuare un frattale che è stato interrotto. Vengono visualizzati i nomi dei frattali incompleti presenti sul dischetto e viene richiesto il nome del frattale da continuare. Premendo solo RETURN si torna al menu principale.

VEDI FRATTALE – visualizza uno dei frattali presenti sul disco.

CAMBIA COLORI – seleziona il colore di sfondo del disegno e il tipo di grafica. La monocromatica è la grafica 9, dove i colori sono le diverse luminosità delle sfondo, mentre il modo multicolor corrisponde alla grafica 11, dove cambiano i colori che però hanno tutti la stessa luminosità dello sfondo.

Andiamo più a fondo: il tracciamento di un frattale viene fatto nella procedura DISEGNO (linee 960-1040), dove viene considerato ogni punto nel campo. C e D contengono il valore del punto in esame riferito a X e Y che sono le coordinate dello schermo. Il frattale viene utilizzato in un'area di 80X180 pixel.

Per ogni punto viene chiamata la procedura PUNTO (linee 800 - 870) che si occupa di stabilire se il punto fa parte dell'insieme. In essa infatti viene eseguita l'iterazione della funzione fino a che il contatore non raggiunga 75 o la dimensione superi 2.

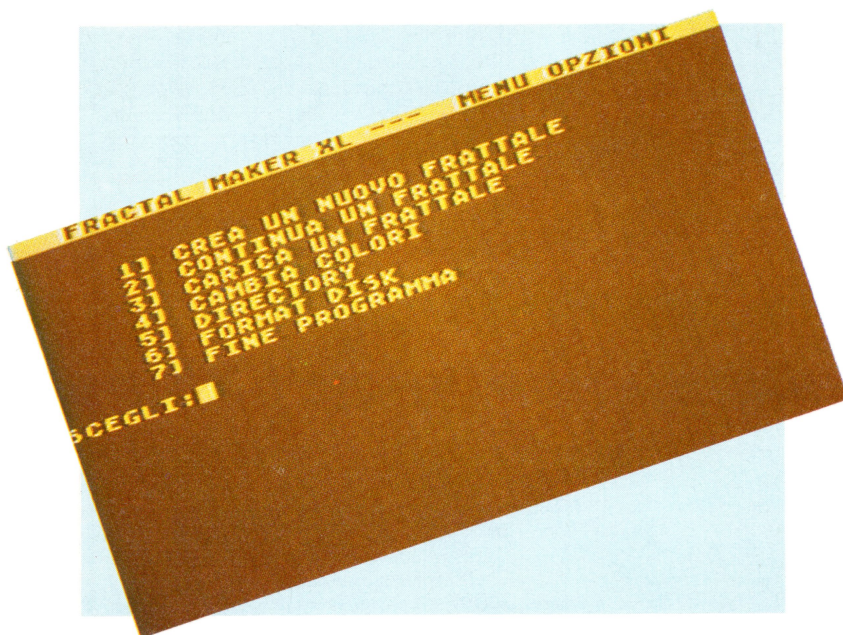
Tutti i punti con dimensione maggiore di 2 sono fuori dall'insieme, perché dopo poche iterazioni andrebbero all'infinito. Tutto l'insieme di Mandelbrot è quindi compreso nel range +-2.

Con i parametri −2, −1.25, 2.5 possiamo visualizzarlo completamente. Il numero di iterazioni massime è fissato a 75, per facilità e brevità di calcolo, ma il programa resta comunque molto lento. La proc. DISEGNO si occupa anche di controllare lo stato dei tasti HELP e SELECT. Premendo HELP si interrompe il

LISTATO FRACTAL MAKER

```
10 REM +--------------------------+
20 REM +   FRACTAL MAKER (XL/XE)   +
30 REM + V 1.4 11,04,87 By EMASOFT +
40 REM +  per La Rivista di ATARI  +
50 REM +--------------------------+
60 POKE 65,0:GOSUB 1300
70 DIM FR$(8),FI$(16),R$(1),DI$(20)
80 DIM CIO$(7):CIO$="hhh":CIO$(4)=CHR$(170):CIO$(5)="LV":CIO$(7)=CHR$(228)
90 GR=9
100 POKE 82,0:GRAPHICS 0:? "   FRACTAL MAKER XL ---  MENU OPZIONI   ":POKE 710,2
24:REM IN INVERSO
110 POKE 82,5:? :? "1] CREA UN NUOVO FRATTALE":? "2] CONTINUA UN FRATTALE":? "3]
 CARICA UN FRATTALE"
120 ? "4] CAMBIA COLORI":? "5] DIRECTORY":? "6] FORMAT DISK":? "7] FINE PROGRAMM
A":POKE 82,0:? :? "SCEGLI:";
130 CLOSE #3:OPEN #3,4,0,"K:":GET #3,K:K=K-48:CLOSE #3:IF K<1 OR K>7 THEN ? CHR$
(253);:GOTO 130
140 ? K:IF K<>7 THEN 170:REM procedura FINISCI ---------------------------------
150 POKE 702,64:PRINT CHR$(253);"SEI SICURO? (S/N)";:INPUT R$:IF NOT (R$="S" OR
R$="N") THEN 150
160 IF R$="S" THEN 710
170 IF K<>5 THEN 270:REM procedura DIRECTORY -----------------------------------
180 CLOSE #2:OPEN #2,6,0,"D:*.*":TRAP 260
190 POKE 710,0:POKE 709,15:? CHR$(125);"        DIRECTORY        ":? :REM IN INVE
RSO
200 ? " L  Filename Ext   Bytes":POKE 82,1:?
210 DD=1:POKE 752,1
220 INPUT #2,DI$:IF DI$(2,2)<>" " THEN POKE 82,0:? :? VAL(DI$(1,3))*128;" Bytes
disponibili":GOTO 260
230 ? DI$(1,2);:POKE 85,4:? DI$(3,10);:POKE 85,12:? ".";DI$(11,13);:DI$=STR$(128
*VAL(DI$(15,17)))
240 POKE 85,24-LEN(DI$):? DI$:DD=DD+1:IF DD=15 THEN POKE 82,0:GOSUB 730:GOTO 190
250 GOTO 220
260 CLOSE #2:GOSUB 730:POKE 82,0
270 IF K<>3 THEN 340:REM procedura CARICA DISEGNO ------------------------------
280 ? ,"FRATTALI DISPONIBILI:":? :OPEN #2,6,0,"D:*.F1":Z=0
290 INPUT #2,DI$:IF DI$(2,2)=" " THEN W=INT(Z/4):Q=Z-(W*4):POSITION 10*Q,15+W:?
DI$(3,10):Z=Z+1:GOTO 290
300 CLOSE #2:IF Z=0 THEN ? "NESSUN FRATTALE SU QUESTO DISCO.":GOSUB 730:GOTO 100
310 TRAP 100:? :? "NOME DEL FRATTALE :";:INPUT FR$:GOSUB 780:IF FR$="" THEN 100
320 GRAPHICS GR:POKE 712,CC:GOSUB 930
330 GOSUB 730
340 IF K<>1 THEN 460:REM procedura CREA FRATTALE -------------------------------
350 TRAP 100:POKE 710,192:? CHR$(125);"   INTRODUZIONE PARAMETRI DI CAMPO    "
:? :REM IN INVERSO
360 ? "ANGOLO SUD-OVEST":? :? "PARTE REALE :";:INPUT ANGR:? :? "PARTE IMM. :";:I
NPUT ANGI:?
370 ? "LATO :";:INPUT LATO:DR=LATO/80:DT=LATO/180:?
380 ? "NOME DEL FRATTALE?":? "(file dest.) :";:INPUT FR$:GOSUB 780:IF FR$="" THE
N 100
390 GRAPHICS GR:POKE 712,CC
400 GOSUB 1140
410 X=0:Y=0:POKE 732,0
420 GOSUB 970
430 GOSUB 890
440 IF X=80 THEN TRAP 450:FI$(LEN(FI$))="2":XIO 33,#3,0,0,FI$
450 GOSUB 730
460 IF K<>2 THEN 600:REM procedura CONTINUA FRATTALE ---------------------------
470 ? ,"FRATTALI DISPONIBILI:":? :OPEN #2,6,0,"D:*.F2":Z=0
480 INPUT #2,DI$:IF DI$(2,2)=" " THEN W=INT(Z/4):Q=Z-(W*4):POSITION 10*Q,15+W:?
DI$(3,10):Z=Z+1:GOTO 480
490 CLOSE #2:IF Z=0 THEN ? "NESSUN FRATTALE INCOMPLETO SUL DISCO.":GOSUB 730:GOT
O 100
500 TRAP 100:? :? "NOME DEL FRATTALE :";:INPUT FR$:IF FR$="" THEN 100
510 GOSUB 780:FI$(LEN(FI$))="2"
520 CLOSE #4:OPEN #4,4,0,FI$
530 INPUT #4;X,Y,DR,DT,ANGR,ANGI
540 CLOSE #4:POKE 732,0:GOSUB 780
550 GRAPHICS GR:POKE 712,CC:GOSUB 930
560 GOSUB 970
570 GOSUB 890
580 IF X=80 THEN TRAP 590:FI$(LEN(FI$))="2":XIO 33,#3,0,0,FI$
590 GOSUB 730
600 IF K<>4 THEN 670:REM procedura CAMBIO COLORI -------------------------------
610 TRAP 610:? :? "COLORE DI SFONDO? (0/255) ";:INPUT C
620 IF C<0 OR C>255 THEN ? CHR$(253);UP$(1,2);:GOTO 610
630 TRAP 33333:CC=C
640 TRAP 640:? :? "TIPO DI GRAFICA?":? "1] MONOCROMATICA":? "2] MULTICOLOR":INPU
T C
650 IF C<1 OR C>2 THEN ? CHR$(253);UP$(1,5);:GOTO 640
660 TRAP 33333:GR=9+2*(C=2)
670 IF K<>6 THEN 700:REM procedura FORMAT DISK ---------------------------------
680 ? :? "INSERISCI UN DISCO VUOTO NEL DRIVE 1.":GOSUB 730
690 TRAP 680:XIO 254,#3,0,0,"D:"
700 GOTO 100
710 GRAPHICS 0:END :REM FINE PROGRAMMA -----------------------------------------
720 REM procedura ATTESA -------------------------------------------------------
730 POKE 752,1:IF PEEK(87)=0 THEN ? :? "PREMI  START  PER CONTINUARE";:REM " STA
RT " IN INVERSO
740 IF PEEK(53279)<>6 THEN 740
750 IF PEEK(87)=0 THEN ? CHR$(156)
760 POKE 752,0:RETURN
770 REM procedura NOMEFILE -----------------------------------------------------
780 FI$="D:":FI$(LEN(FI$)+1)=FR$:FI$(LEN(FI$)+1)=".F1"
790 RETURN
800 REM procedura PUNTO --------------------------------------------------------
810 DI=0:AP=0:BP=0:CO=0
820 CO=CO+1
830 NA=AP*AP-BP*BP:NB=2*AP*BP
840 AP=NA+C:BP=NB+D
850 DI=SQR(AP*AP+BP*BP)
860 IF NOT (CO=75 OR DI>2) THEN 820
870 RETURN
```

```
880 REM procedura SALVA PICTURE ----------------------------------------
890 POKE 559,34:TRAP 910:CLOSE #1:OPEN #1,8,0,FI$
900 IOCB=1:TYPE=11:GOSUB 1230:XX=USR(ADR(CIO$),IOCB*16)
910 CLOSE #1:RETURN
920 REM procedura CARICA PICTURE ---------------------------------------
930 TRAP 950:CLOSE #1:OPEN #1,4,0,FI$
940 IOCB=1:TYPE=7:GOSUB 1230:XX=USR(ADR(CIO$),IOCB*16)
950 CLOSE #1:RETURN
960 REM procedura DISEGNO -------------------------------------------
970 IF PEEK(53279)=5 THEN POKE 559,34-PEEK(559)
980 C=ANGR+X*DR:D=ANGI+Y*DT
990 GOSUB 810:COLOR CO/5:PLOT X,Y
1000 Y=Y+1:IF Y=180 THEN Y=0:X=X+1
1010 IF PEEK(732) THEN GOSUB 1060:GOTO 1040
1020 IF X=80 THEN 1040
1030 GOTO 970
1040 RETURN
1050 REM procedura NONFINE ---------------------------------------------
1060 GOSUB 780
1070 FI$(LEN(FI$))="2"
1080 TRAP 1100:CLOSE #4:OPEN #4,8,0,FI$
1090 ? #4;X;",";Y;",";DR;",";DT;",";ANGR;",";ANGI
1100 CLOSE #4
1110 GOSUB 780
1120 RETURN
1130 REM procedura GRIGLIA ---------------------------------------------
1140 COLOR 15
1150 FOR XX=0 TO 40 STEP 5
1160   FOR YY=0 TO 90 STEP 10
1170     PLOT XX,YY:PLOT XX,179-YY
1180     PLOT 79-XX,YY:PLOT 79-XX,179-YY
1190   NEXT YY
1200 NEXT XX
1210 RETURN
1220 REM setta parametri i/o-picture -----------------------------------
1230 BLOCK=832+IOCB*16:LN=7680
1240 L=LN:H=INT(L/256):L=L-256*H
1250 POKE BLOCK+2,TYPE
1260 POKE BLOCK+5,PEEK(89):POKE BLOCK+4,PEEK(88)
1270 POKE BLOCK+9,H:POKE BLOCK+8,L
1280 RETURN
1290 REM procedura TITOLO-----------------------------------------------
1300 POKE 82,0:GRAPHICS 2:POKE 752,1:POSITION 6,4:? #6;"fractal":POSITION 7,6:?
#6;"MAKER"
1310 COLOR 139:PLOT 0,0:DRAWTO 19,0:DRAWTO 19,9:DRAWTO 0,9:DRAWTO 0,0
1320 POKE 710,0:POKE 709,30:POKE 708,154
1330 DIM UP$(6):UP$(1)=CHR$(28):UP$(6)=CHR$(28):UP$(2)=UP$
1340 ? CHR$(125):? "      PREMI  START  PER INIZIARE.":REM " START " IN INVERSO
1350 POKE 708,PEEK(20):POKE 711,PEEK(19):POKE 709,255-PEEK(20)
1360 IF PEEK(53279)<>6 THEN 1350
1370 RETURN
```

In apertura l'immagine di un frattale. Qui sopra il menu principale di Fractal Maker

disegno, che viene salvato incompleto, sarà possibile continuarlo in un secondo tempo con l'opzione CONTINUA. Il tasto SELECT funziona da toggle per lo schermo, cioè serve a abilitarlo o a disabitarlo. Con lo schermo disabilitato il computer lavora più velocemente.
Poiché il controllo sul tasto SELECT viene effettuato solo una volta per ogni punto, può capitare che dobbiate tenere permuto il tasto per qualche secondo prima che inverta lo schermo. Per il tasto HELP invece questo problema non sussiste perché la locazione 732 non viene riazzerata quando viene rilasciato il tasto. Basta premerlo per un istante, anche se il salvataggio verrà fatto qualche secondo dopo.
Il colore del punto viene attribuito in funzione del valore del contatore. I punti esterni sono neri, e sono quelli più rapidi da calcolare.
Un'immagine viene salvata in formato non compatto, a 62 settori, quindi compatibile con i maggiori programmi grafici. L'immagine viene salvata in un file con l'estensione F1. Inoltre se il frattale è incompleto viene creato un frattale con l'estensione F2 che contiene i parametri di quando si è interrotto il disegno.
Le operazioni di I/O vengono effettuate usando una routine che utilizza una chiamata al CIO, che potrebbe essere il soggetto di un prossimo articolo.
La procedura GRIGLIA traccia una griglia punteggiata sullo schermo per poter vedere a che punto dello schermo è arrivata l'esecuzione.
Dopo che un'immagine è stata completata oppure è stata salvata con HELP o richiamata con l'opzione 3, il programma

CODICI PROOF FRACTAL MAKER

LINE#	CODE	LINE#	CODE	LINE#	CODE
10	0222	20	7871	30	9100
40	8067	50	0226	60	5939
70	1517	80	7050	90	1708
100	2777	110	5248	120	2725
130	4755	140	1439	150	3018
160	4070	170	0629	180	7040
190	9369	200	8491	210	0669
220	6655	230	5326	240	4533
250	4391	260	1055	270	3555
280	8835	290	3154	300	7998
310	9370	320	3710	330	5270
340	2678	350	5098	360	5510
370	5884	380	6475	390	3505
400	5899	410	4521	420	5339
430	5354	440	3067	450	5275
460	9437	470	8888	480	3255
490	6297	500	3862	510	6715
520	2285	530	4930	540	3379
550	3718	560	5348	570	5363
580	3195	590	5284	600	3008
610	1857	620	7263	630	1235
640	8189	650	1498	660	0664
670	0912	680	8635	690	7010
700	4354	710	1501	720	9606
730	6979	740	6419	750	3045
760	5411	770	0882	780	0252
790	4051	800	9553	810	6953
820	3880	830	7470	840	2065
850	7323	860	6537	870	4048
880	4793	890	9803	900	5351
910	3033	920	5043	930	9402
940	2256	950	3041	960	3330
970	5124	980	5222	990	7029

CODICI FRACTAL PARTE 2

LINE#	CODE	LINE#	CODE	LINE#	CODE
1000	6793	1010	5975	1020	4142
1030	5243	1040	4743	1050	1146
1060	6139	1070	2939	1080	3845
1090	0064	1100	5094	1110	6126
1120	4739	1130	0821	1140	5378
1150	1059	1160	2365	1170	1791
1180	3785	1190	5448	1200	5400
1210	4738	1220	1481	1230	7701
1240	1116	1250	6866	1260	2167
1270	7878	1280	4759	1290	1319
1300	4165	1310	0653	1320	6077
1330	6841	1340	5457	1350	2310
1360	9653	1370	4758		

LISTING 2

```
45 REM * LINEE PER DOPPIA PRECIS.  *
860 IF NOT (CO=150 OR DI>2) THEN 820
990 GOSUB 810:COLOR CO/10:PLOT X,Y
```

LINE#	CODE	LINE#	CODE	LINE#	CODE
45	1068	860	8345	990	8763

LISTING 3

```
45 REM * LINEE PER PIXEL QUADRATI  *
280 ? ,"FRATTALI DISPONIBILI:":? :OPEN #2,6,0,"D:*.F3":Z=0
370 ? "LATO :";:INPUT LATO:DT=LATO/80:?
440 IF X=80 THEN TRAP 450:FI$(LEN(FI$))="4":XIO 33,#3,0,0,FI$
470 ? ,"FRATTALI DISPONIBILI:":? :OPEN #2,6,0,"D:*.F4":Z=0
510 GOSUB 780:FI$(LEN(FI$))="4"
530 INPUT #4;X,Y,DT,ANGR,ANGI
580 IF X=80 THEN TRAP 590:FI$(LEN(FI$))="4":XIO 33,#3,0,0,FI$
780 FI$="D:":FI$(3)=FR$:FI$(LEN(FI$)+1)=".F3"
860 IF NOT (CO=150 OR DI>2) THEN 820
980 C=ANGR+X*DT:D=ANGI+Y*DT
990 GOSUB 810:COLOR CO/10:PLOT X,2*Y:PLOT X,2*Y+1
1000 Y=Y+1:IF Y=80 THEN Y=0:X=X+1
1070 FI$(LEN(FI$))="4"
1090 ? #4;X;",";Y;",";DT;",";ANGR;",";ANGI
1160 FOR YY=0 TO 80 STEP 10
1170   PLOT XX,YY:PLOT XX,159-YY
1180   PLOT 79-XX,YY:PLOT 79-XX,159-YY
```

LINE#	CODE	LINE#	CODE	LINE#	CODE
45	2381	280	8941	370	1030
440	3151	470	8994	510	6775
530	8799	580	3279	780	5337
860	8345	980	5252	990	6759
1000	4904	1070	2981	1090	1957
1160	2347	1170	1739	1180	3721

LISTING 4

```
10 GRAPHICS 0:? "sto controllando le linee DATA...":CK=0
20 FOR T=1 TO 139:READ A:CK=CK+A:NEXT T
30 IF CK<>13328 THEN ? CHR$(253);"ERRORE NEI DATA!!":END
40 RESTORE :OPEN #1,8,0,"D:AUTORUN.SYS"
50 ? "DATA Ok.":? "Sto creando il file AUTORUN.SYS"
60 FOR T=1 TO 139
70   READ A:PUT #1,A
80 NEXT T
90 CLOSE #1:? "AUTORUN.SYS creato!"
100 END
110 DATA 255,255,160,6,162,6,76,175,6,175,6,251,6,160,11,185
120 DATA 0,228,153,163,6,136,16,247,169,222,141,167,6,169,6,141
130 DATA 168,6,172,170,6,174,169,6,232,208,1,200,142,246,6,140
140 DATA 247,6,169,163,141,33,3,169,6,141,34,3,96,172,0,6
150 DATA 208,10,169,0,141,33,3,169,228,141,34,3,185,1,6,206
160 DATA 0,6,72,32,251,6,104,160,1,96,253,6,255,6,108,250
170 DATA 191,68,2,68,2,0,9,0,9,0,1,226,2,227,2,160
180 DATA 6,224,2,225,2,253,6,0,6,15,6,14,155,73,76,65
190 DATA 84,84,65,82,70,58,68,34,78,85,82
```

LINE#	CODE	LINE#	CODE	LINE#	CODE
10	1331	20	0118	30	5196
40	0720	50	3242	60	8703
70	9874	80	3033	90	6892
100	1706	110	0888	120	0380
130	6243	140	1811	150	7298
160	1724	170	2406	180	1040
190	3880				

si aspetta che voi premiate START prima di tornare al menu. Fate attenzione che un salvataggio su disco termina quando il drive ha finito di girare. Il suono normale del drive è stato disinserito. Quando un frattale viene completato, viene salvato automaticamente, e viene cancellato l'eventuale file dei dati se era stato interrotto.

I disegni migliori sono quelli che ingrandiscono particolari del margine: provate ad esempio con questi valori: −0.75, 0.25, 0.1 oppure −1, −2.5, 0.1 e 0.02, −0.79, 0.05. Non preoccupartevi se le prime volte che introdurrete valori a caso non vedrete niente: non è facile trovare subito zone "belle".

E ora qualche consiglio per gli smanettoni esigenti.

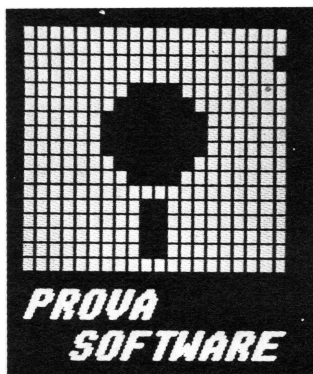
Probabilmente 75 iterazioni lasciano fuori una zona di margine interessante. Per migliorare la precisione (a scapito della velocità) si possono modificare le linee del listato 2. Se poi non vi piace il fatto di avere una zona quadrata rappresentata con pixel orizzontali, modificate le linee del listato 3 in modo da avere una risoluzione di 80X80 dove vengono accesi due pixel alla volta, così, da avere i punti un po' più quadrati. Questa versione è già predisposta a doppia precisione, e cambia le estensioni dei files (F3 = disegno, F4 = dati), così da non creare conflitti tra i disegni creati dalle due versioni.

Per chi ama le sfide difficili consiglio di provare e adattare il programma alla grafica 15.

Vista l'inevitabile lentezza del programma (se trovate un frattale da meno di 3 ore fatecelo sapere!), consigliamo a chi lo possiede di compilare il programma.

Stavamo per dimenticarci del listato 4. Questo serve a creare su un disco un file AUTORUN.SYS che carica e esegue un file in basic chiamato FRATTALI. Quindi se salvate il programma con il nome FRATTALI e se il disco contiene questo Autorun (e il DOS), il programma viene caricato all'atto dell'accensione (senza premere OPTION).

ELENCO VARIABILI

FR$ = Nome del frattale.
FI$ = Nome del file
CIO$ = Routine CIO
GR = Modo grafico .
ANGR,ANGI = Numero complesso, angolo del campo.
LATO = Lato del campo complesso.
DR,DT = Fattori di scala assi x,y
X,Y = Coordinate punti sullo schermo
C,D = Numero complesso corrispondente a X,Y,
AP,BP = Numero complesso Z.
DI = Dimensione di Z.
CO = Contatore di iterazioni.
IOCB = Canale utilizzato dal CID
TYPE = Operazione CIO (7=READ.11=WRITE).
BLOCK = Indirizzo di inizio del canale CIO.
LN = Numero di bytes di una immagine

DBMan

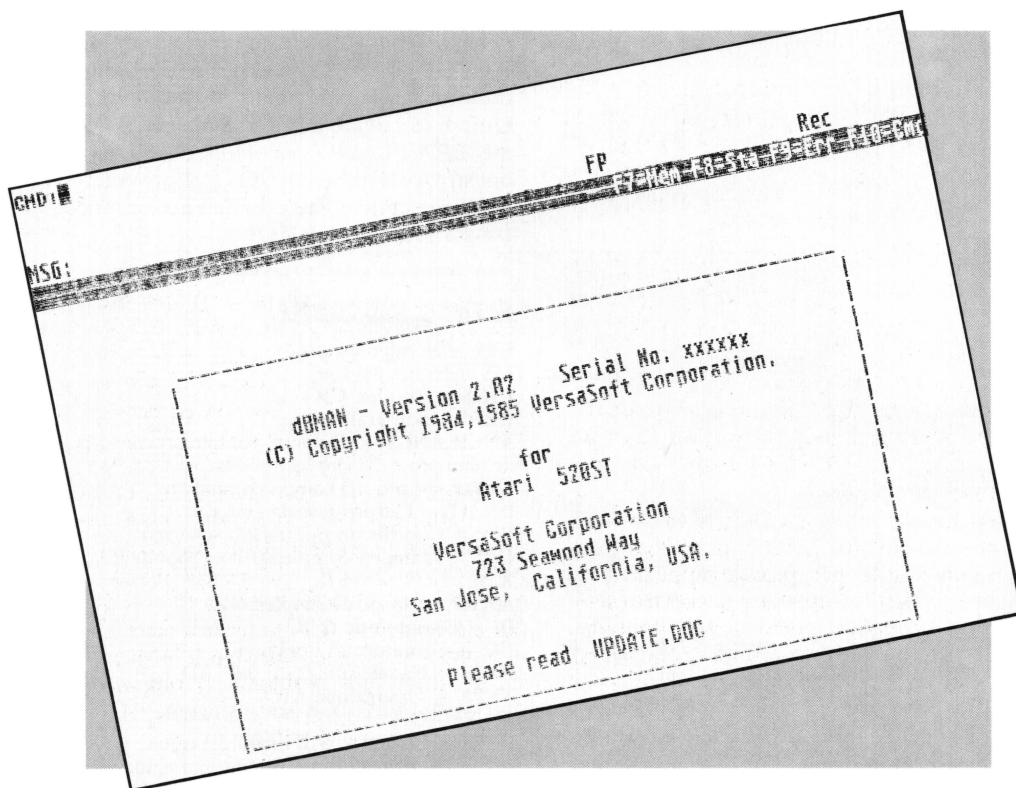
DI MATTEO PRINETTI

UN LINGUAGGIO DI PROGRAMMAZIONE PER LA GESTIONE DEI DATI

Per l'Atari ST ci sono in giro molti Database. Dal DB Master al CARD File ecc... La maggior parte di questi permettono di definire un archivio e di effettuare semplici operazioni su di esso (ordinamenti, ricerche, generazioni di tabulati ecc.) ma risentono tutti delle limitazioni imposte principalmente dal fatto che sono programmi finiti e non linguaggi di programmazione. Ci sono poi programmi che si distaccano dalla categoria per diventare veri e propri linguaggi di programmazione. È il caso di Holmes & Duckworth ma soprattutto di DBMan, in pratica la versione per Atari St del programma DB III prodotto per MS DOS dalla Ashon Tate.
Dire in poche righe cosa rappresenti il DB III per la programmazione gestionale è impossibile. Si tratta di uno di quegli standard che sono oramai talmente attecchiti nelle usanze dei programmatori che senza di esso non possono più sopravvivere. In effetti il suo grande pregio (e che si trasferisce di pari passo nel DBMan) è che può essere usato sia dal programmatore che dall'utente finale con estrema semplicità, e che ha una gestione dell'I/O (sia dai files che da tastiera) veramente ottima.
Di contro è un prodotto "dedicato". Non ci sono la maggior parte delle funzioni matematiche estese come la notazione esponenziale, le funzioni trigonometriche, completamente assenti funzioni grafiche evolute. Insomma DBMan è ottimo dovunque sia presente una problematica di tipo aziendale, vedi contabilità, gestione ordini/magazzino, ecc. e non in ambito tecnico-scientifico con il quale va poco d'accordo.
Vediamo di entrare però nel dettaglio. Innanzitutto la confezione. DBMan risiede su tre dischi tutti non protetti e necessita di 400K di Ram libera (chi avesse uno dei primi 520 ST senza le ROM del TOS è pregato di comprarle), ma il disco fisso è molto consigliato, dato che senza di esso i programmi girano molto lentamente a causa dei continui accessi al disco. Un bel manuale di 200 pagine chiaramente in inglese completa il tutto, fortunatamente si può ricorrere a uno dei tanti manuali tecnici per il DB III per MS DOS editi in italiano dal quale non risulta difficile estrapolare la spiegazione della maggior parte dei comandi.
Una volta caricato, il dischetto si può togliere per inserire il nostro disco. Si rileva che non sempre il programma funziona al primo colpo, nel senso che bisogna resettare la macchina e ricaricare. Questo inconveniente è stato riscontrato soprattutto sui vecchi 520 ST. Dopodiché si può cominciare a lavorare.

CARATTERISTICHE TECNICHE

Ogni file può avere fino a 128 campi per un totale di 4000 caratteri per record. Contemporaneamente si possono tenere

CMD:
MSG:
dBMAN - Version 2.02 Serial No. XXXXXX
(C) Copyright 1984,1985 VersaSoft Corporation.
for
Atari 520ST
VersaSoft Corporation
723 Seawood Way
San Jose, California, USA.
Please read UPDATE.DOC

La schermata iniziale di D.B. Man.

aperti 10 files. Ci sono quattro tipi di dati e cioè stringhe di caratteri (fino a 254 caratteri per stringa), date, variabili logiche ("vero" e "falso") e numeri, con 15 cifre significative.
Con questi "spazi di manovra" si riescono a coprire la maggior parte delle applicazioni. Come accennato DBMan può essere usato in due modi. Da utente o da programmatore. In pratica si può scrivere un programma in DBMan che non è altro che la sequenza dei comandi dati da tastiera integrati con comandi per la gestione del flusso del programma (IF,DO..WHILE, ecc...).
Se si usa DBMan in modo utente non si

Infatti è sempre difficile controllare che un Input numerico sia tale, compreso in un range, o che uno alfanumerico sia tutto maiuscolo o allineato a sinistra, o che una data sia "data". Il DBMan permette una completa gestione dell'Input che soddisfa tutte le esigenze, ed è un vero sollievo non dover mettersi a pensare sempre se il dato sarà esatto, controllare che sia numerico, che non vengano usati tasti di direzione (freccette ecc...). Solo questa parte giustifica il prodotto.
Se nel vostro programma usate i file, andate incontro ai soliti macelli. Definire il record, la struttura, su cosa fare l'indice, creare il file indice, tenere aggiornati en-

fare con relativamente poca fatica.
Oltre ai due dischi ne viene fornito un terzo con su scritto "Compilatore" a mano, senza manuale, che non so se è presente anche nella confezione in commercio. Comunque non compila un tubo, si limita a comprimere i programmi che così girano un po' più in fretta e non sono visibili con l'editor, dimodoché risultano "protetti".

CONCLUSIONI

DBman è sicuramente un programma utile, ma dato che viene venduto dalla Atari sarebbe auspicabile un manuale in italiano. Inoltre il programma gira più

La definizione di un archivio di dati.

La visione complessiva di tutti i dati contenuti in un report si ottiene con il comando Display All.

hanno molti problemi. I guai iniziano se si scrive un programma e si prova ad eseguirlo. Infatti in DBMan non c'è un vero e proprio editor, per cui bisogna uscire da DBMan, caricare l'editor, modificare il programma, ricaricare DBMan e riprovare ad eseguirlo. Ci sono dei tasti di debugging ma non li ritengo molto utili, personalmente preferisco avere l'editor "in linea" e rinunciare ai tasti di debug.
Comunque i presupposti per scrivere buoni programmi ci sono tutti, soprattutto sono presenti tutte le facilitazioni previste dal DB III per la gestione degli indici e dell'input dallo schermo. Se pensate attentamente alla struttura di un programma vi accorgete che i controlli sull'input rappresentano il 50% del lavoro.

trambi in contemporanea. Con DBman vi dimenticate tutto questo perché il programma fa tutto in automatico. Si possono definire file, indici multipli, cambiare la struttura di un record senza perdere i dati in esso contenuti, insomma questa altra parte di DBman giustifica completamente il prodotto.
C'è una utility presente per creare dei tabulati, invero molto semplice ma comunque utile per piccole applicazioni; per prospetti complessi è necessario scrivere un programma che genera il tabulato. Manca l'utility presente nel DB III per la stampa delle etichette. Infine per chi desideri vedere qualcosa in DBMan, c'è un piccolo programma di Mailing List che però rende bene l'idea di cosa si può

lentamente che il DB III e la compatibilità è a volte complessa da gestire. Si apre comunque una grande strada per i programmatori, e cioè di trasferire la marea di Software scritto in DB III sotto ST. Quanto ciò sia possibile è da verificare. Viene infine spontanea una domanda. Quanto è utile DBMan al nostro amico "smanettone"? Poco o niente, infatti è ormai da tempo consolidata la constatazione che per tenere la lista degli indirizzi è meglio una rubrica e relativa matita. Se poi c'è qualcuno (c'è?) che utilizza l'ST in ambito aziendale, DBMan è la risposta a tutti i suoi problemi. Il rapporto prestazioni/prezzo è soddisfacente (DB III costa 10 volte di più!).

L'INFORMAZIONE CORRE SUL FILO

LA TRASMISSIONE DEI DATI VIA TELEFONO STA VIVENDO UN MOMENTO DI ECCEZIONALE VIGORE. GLI UTENTI DI ATARI ST NON POSSONO STARE ALLA FINESTRA: ECCO A VOI QUATTRO PROGRAMMI DI COMUNICAZIONE PER IL VOSTRO COMPUTER.

DI PAOLO GALVANI

Fino a non molto tempo fa, acquistare un modem significava investire ingenti capitali e rischiare di trovarsi con un oggetto poco funzionale. Ora si trovano in commercio apparecchi affidabili e di costo contenuto già a partire dalle 150.000 lire. Così la comunicazione tramite linee telefoniche si espande senza sosta, e non passa giorno senza che nasca qualche nuova banca dati.
L'errore che comunemente si tende a fare è quello di scegliere con molta attenzione il modem (nome derivante dalle parole "modulatore-demodulatore") e lasciare al caso, o quasi, la scelta del software. Eppure, al pari dei modem, i programmi di comunicazione possono variare notevolmente da uno all'altro, tanto che, senza far attenzione, potremmo acquistare software completamente inadatto al nostro lavoro.
Abbiamo messo faccia a faccia quattro programmi di comunicazione di vario tipo, dai semplici pacchetti di comunicazione agli emulatori di terminale, per verificare quali sono le principali caratteristiche per ognuno di essi.
Ecco allora K-Komm 2, Flash, PC/InterComm e FaSTcom affiancati per la nostra prova.
Prima di passare al confronto diretto vorremmo però soffermarci sui parametri di trasmissione dati oggi utilizzabili.

I PARAMETRI DI TRASMISSIONE

Il computer con il quale intendiamo collegarci viene definito "host", ospite, e per poter dialogare con lui dovremo adeguare la nostra macchina al suo standard. In primo luogo va definita la velocità di trasferimento dati, misurata in baud. Questa è generalmente variabile tra i 50 e i 9600 baud; le normali linee telefoniche non riescono a inviare un segnale perfetto ad altissima velocità perché la trasmissione analogica porta ad avere dei disturbi di linea. Quindi è chiaro che per non perdere parti del segnale la trasmissione va effettuata ad una velocità piuttosto lenta, di solito 300 o 1200 baud.
I dati, per essere trasmessi, vengono inviati divisi a "gruppetti" che possono es-

sere formati da sette oppure otto bit. È perciò necessario selezionarne la giusta quantità a seconda di quanto usato dal computer remoto.
A questi bit, in alcuni casi, va aggiunto il bit di parità: se il gruppetto di sette (o otto) bit contiene in maggiore numero degli "0", allora dovremo scegliere "pari", se al contrario sono gli "1" a prevalere selezioneremo "dispari".
Un ulteriore controllo va aggiunto per segnalare all'host computer che il gruppetto è stato interamente trasmesso. Per farlo inseriremo, a seconda dei casi, uno o due bit denominati "di stop".
Altra selezione da effettuare è quella tra Half e Full Duplex. Quando trasmettiamo, abbiamo la possibilità di vedere sul nostro monitor i dati nel momento in cui li inviamo (Half Duplex) oppure di riceverli (e vederli) di ritorno dall'host, in modo da controllare l'assenza di errori (Full Duplex).
Queste selezioni vanno fatte se possediamo un pacchetto di comunicazione. Spesso il software ci mette però a disposizione dei protocolli di trasmissione, ovvero una serie di parametri fissi e standard con i quali settare la nostra RS232. L'emulazione di un terminale ci consente invece di trasformare il nostro computer in una macchina dedita esclusivamente alla trasmissione/ricezione dei dati.

AM COMPUTER TECHNOLOGY FASTCOM

Il primo programma preso in considerazione è FaSTcom. Studiato appositamente per i neofiti è essenziale nelle prestazioni e intuitivo nell'uso.
La schermata di presentazione ispira immediatamente un senso di confidenza: incolonnate ordinatamente sul lato sinistro vi sono, tranquillizzanti, diverse icone, mentre nella barra orizzontale superiore trovano posto i ben conosciuti menu pull-down.
L'interfaccia utente è quella consueta, cosicché l'utilizzatore si trova a suo completo agio usando il familiare mouse.
ICONA TELEFONO - Raffigura un'apparecchio telefonico e selezionandola con il mouse appare una finestra contenente una rubrica telefonica. Qui compaiono tutti i nominativi con i relativi numeri di telefono che in precedenza abbiamo memorizzato. Anche se non sono visibili, ad ogni nome sono associati i relativi parametri.
Nella stessa finestra vi sono tre tasti: "Dial" – per chiamare –, "Redial" – per richiamare – e "Local" – per collegarsi ad un host direttamente senza connessione telefonica.
ICONA TELEPHONE BOOK - Si trova appena sotto all'altra e viene utilizzata per inserire i nuovi nominativi. Qui dobbiamo settare i parametri che verranno poi memorizzati nella prima icona.
FaSTcom possiede solo il protocollo TTY, che è anche il più diffuso e il più semplice. La velocità di trasmissione è variabile sia per l'invio dei dati, sia per la ricezione. Le frequenze impostabili sono 75, 110, 150, 300, 600, 1200, 2400, 4800 e 9600 Baud.
ICONA MODEM - Contiene l'elenco dei più diffusi modem con i relativi parametri. Se il modem in vostro possesso non compare, ma questo è molto difficile, potrete definirlo voi stessi.
Prima di effettuare una chiamata dovrete quindi selezionare il modello del vostro apparecchio da questo elenco.
ICONA TESTO - Come molti altri programmi di comunicazione, anche FaSTcom offre la possibilità di realizzare testi con stili differenti da quello normale. Con questa opzione potete scegliere tra i caratteri piani, in grassetto, italici, sottolineati, contornati e ombreggiati.
Oltre a questo, se avete anche un monitor a colori, vi viene offerta la possibilità di variare il colore dello schermo. Le tonalità disponibili sono il rosso, il blu, il verde, il magenta, il ciano e il giallo.
ICONA STATISTICHE - Memorizza automaticamente tutte le chiamate indicando il nominativo, con relativo numero telefonico, con il quale siamo stati collegati, la data e l'ora del collegamento e la sua durata.
ICONE DOCUMENTO - Servono a definire il tipo di file da trasmettere/ricevere. Questi possono essere testi, immagini (che possono essere ricevute ma non create con FaSTcom), messaggi e sequenze logiche.
ICONA HELP - In ogni momento possiamo caricare un file di aiuto relativo alle operazioni che svolgiamo. Per farlo è sufficiente selezionare questa icona e indicare quale file intendiamo visionare.

Grazie alla serie di icone poste sul lato sinistro e all'uso del mouse, FaSTcom risulta essere il più intuitivo tra i pacchetti di comunicazione provati.

I MENU PULL-DOWN

Come detto, oltre alle icone abbiamo anche i classici menu a discesa. In alcuni casi essi riportano le funzioni delle icone, in altri offrono altre opzioni.
MENU FILE - Ovviamente contiene tutti i comandi per il trattamento dei file. Troviamo quindi "New", "Open", "Close", "Close all", "Save" e "Save as". Oltre a questi vi sono anche i comandi per trasfe-

rire i dati, ovvero "Send file" (per inviarli), "Receive file" (per riceverli) e "Send Message" (per inviare un messaggio).

MENU EDIT - È il menu che permette il trattamento dei testi. Abbiamo quindi le solite funzioni di taglia, copia e incolla nonché "Undo", "Show clipboard" e "Find and replace". In più troviamo "Select all", "Set Text Width" (per stabilire la larghezza del testo) e "Save with CR" (per salvare i dati con i Carriage Return, ovvero gli "a capo").

MENU COMMANDS - In questo caso il menu riporta le stesse funzioni delle icone.

MENU THINGS - In questo menu troviamo alcune interessanti opzioni. Ad esempio la "Tidy windows", per pulire le finestre, o l'"Auto grab", ovvero la funzione che registra automaticamente i file ricevuti. Sempre nello stesso menu abbiamo il comando di definizione delle Macro e dei tasti funzione.

Le altre funzioni sono "Hangup", per terminare la comunicazione, "Clear rec text", per cancellare il testo registrato, "Function key setup", per assegnare stringhe di caratteri ai tasti funzione, e "Send selected text", per inviare la parte di testo selezionata.

IMPRESSIONI D'USO

FaSTcom è un programma di semplicissimo utilizzo con interessanti funzioni adattissimo per chi si avvicina alla trasmissione dati per la prima volta. Qualsiasi difficoltà può comunque venire superata con la funzione Help, che dà, in ogni momento, utili indicazioni su tutte le funzioni. Grazie a queste sue prerogative, FaSTcom si propone sicuramente come uno tra i più validi programmi di comunicazione della sua categoria.

ANTIC FLASH

Con questo pacchetto entriamo nei programmi un po' più complicati nell'uso, ma che offrono anche una maggiore versatilità, in questo caso il trattamento di blocchi.

Con la maggiore sofisticazione spariscono le icone e rimangono semplicemente i menu a discesa.

L'utilizzo comunque non è proibitivo, e con un minimo di applicazione il tempo di apprendimento è molto breve. Lo svantaggio rispetto a FaSTcom è che Flash è molto meno istintivo.

MENU FILE

In questo caso i comandi non sono molti, solamente quelli essenziali per il trattamento dei dati. Ci sono "Load", "Save", "Delete" e "Merge", che serve a caricare una serie di nominativi tratti, per esempio, da un database.

Molto comoda in Flash la finestra che riassume tutte le caratteristiche impostate per la comunicazione. Ben strutturata e completa la funzione di help.

MENU EDIT

Ritroviamo tutte le funzioni di settaggio dei parametri di trasmissione dei dati. Esaminiamo con ordine le possibilità che ci vengono offerte da questo menu.

FUNCTION KEYS - Permette di assegnare una stringa di caratteri oppure un comando ad ogni tasto funzione. È spesso utile per avere sempre a disposizione la propria pass-word o i codici di altri utenti cosicché non saremo costretti a digitarli ogni volta.

DIAL DIRECTORY - È la rubrica contenente i numeri dei nominativi memorizzati.

XMODEM PARAMETERS - Mentre FaSTcom utilizza il protocollo TTY, Flash adotta l'Xmodem. Con questa opzione è possibile apportarvi modifiche.

RS232 CONFIGURATION - Da questo punto di vista Flash è meno versatile di FaSTcom, non potendo selezionare velocità basse come 75, 110 o 150 Baud. Le possibilità offerte sono: 300, 600, 1200, 4800 e 9600 Baud.

Naturalmente presenti la dimensione dei gruppi di bit (7 o 8), la parità e i bit di stop (1 o 2).

ASCII UL-DL - Contiene le opzioni per il trasferimento dati. Tra le altre cose permette di avere una pausa al termine di ogni linea o di stabilire il carattere di prompt.

MENU UPLOAD/DOWNLOAD

Sono i menu che consentono la trasmissione e la ricezione dei dati. Upload offre diverse possibilità, ovvero:

File Xmodem - Trasmette un file in protocollo Xmodem.

File ASCII - Trasmette un file trasformandolo in codice ASCII.

Capture Xmodem - Trasmette un file in protocollo Xmodem lasciando invariati i codici di controllo del testo.

Capture ASCII - Trasmette un file in codice ASCII con i codici di controllo.

Block Xmodem - Trasmette un blocco precedentemente selezionato in protocollo Xmodem.

Block ASCII - Trasmette un blocco trasformandolo in codice ASCII.

Il menu Download consente di ricevere file in protocollo Xmodem ed in codice ASCII.

MENU BLOCK

Abbiamo parlato di trasmissione di blocchi, ma non abbiamo ancora detto come crearli. All'uopo vi è il menu Block, che serve proprio a questo: al trattamento e alla creazione dei blocchi.
MARK STAR - Indica il punto di inizio del blocco.
MARK END - Indica la fine del blocco. Un blocco è costituito dalla parte di testo compresa tra il Mark start e il Mark end.

CLEAR MARKS - Elimina i limiti del blocco.
FIND START - A volte risulta difficoltoso risalire all'inizio di un blocco. Con questa opzione Flash lo trova immediatamente.
DELETE BLOCK - Cancella un blocco.
SAVE, PRINT, APPEND BLOCK - Sono i tre comandi che permettono di salvare, stampare e aggiungere blocchi.

MENU OPTIONS

Nel menu delle opzioni troviamo alcune funzioni che risultano utili a velocizzare il lavoro e, come nel caso di "Status", ci riepilogano il nostro stato.
HELP e EDITING HELP - Editing Help accede ad una finestra che ricorda qual è la funzione di alcuni dei principali tasti, Help permette invece di ottenere un aiuto relativo ai principali comandi.
SET CLOCK - L'Atari non possiede l'orologio autoalimentato come nel caso del Macintosh, per cui ogni volta va regolato.
STATUS - Dà utili informazioni sullo stato della porta RS232, sul protocollo di trasmissione e sul modem.
FIND STRING - Trova una stringa di caratteri all'interno del testo.
PRINT CAPTURE E CLEAR CAPTURE - Rispettivamente stampa e cancella i Capture File.

IMPRESSIONI D'USO

Rispetto a FaSTcom, Flash è un po' più ruvido nell'uso, ovvero richiede una maggiore attenzione almeno le prime volte. La mancanza di un punto di riferimento semplice e chiaro come le icone ha il suo peso. Nonostante ciò, non appena si è presa un po' di confidenza con il programma tutto fila liscio e il lavoro procede speditamente.
A parte la differenza nel protocollo di trasmissione adottato dai due programmi, Flash si differenzia per la comoda possibilità di manipolare blocchi di testo.

KUMA K-COMM 2

Con questo programma della Kuma Computers, distribuito nel nostro paese dalla Atari Italia, ci spostiamo su programmi di ben altra categoria: gli emulatori.
K-Comm 2 è in grado di emulare senza difficoltà un semplice terminale Teletype, ma anche i più complessi, e molto più diffusi, VT52 e VT100. Grazie all'opzione "Configure terminal" è anche possibile emulare altri terminali.
Ovviamente l'uso della tastiera, ed in particolare del tastierino numerico, sarà leggermente differente dal solito, ragion per cui è consigliabile per i primi tempi avere sempre sottomano il manuale.
Anche in questo programma non vi sono icone, ma solamente menu a discesa.
Insieme a K-Comm 2, emulatore di terminali, viene fornito anche K-Vievw, pacchetto per la comunicazione, di cui in questo articolo non ci occupiamo. La presenza di questo programma va comunque tenuta in considerazione al momento dell'acquisto.

Con K-Comm 2 si entra nel campo degli emulatori di terminale, e di conseguenza aumentano le difficoltà d'uso. Da notare la comodità di poter variare a piacimento il layout della tastiera.

MENU FILE

Per gestire i file troviamo diverse opzioni, dalle semplici "Open" e "Close" alle più complesse operazioni di trasferimento.
RECEIVE, PAUSE, CLOSE - Sono i comandi usati per ricevere i dati dall'host computer. "Pause" consente di fermare momentaneamente la ricezione.
SEND - Spedisce i dati all'elaboratore collegato.
TAKE SNAPSHOT - Selezionando questa opzione, K-Comm 2 "fotografa" lo schermo e ne memorizza l'immagine in uno speciale buffer. Per visionare quanto memorizzato basterà usare l'opzione "View snapshot". Se selezioniamo due volte "Take snapshot", la prima schermata verrà cancellata dal buffer.
XMODEM - È il classico protocollo già adottato anche da Flash.

MENU PRINTER

È il menu che gestisce il funzionamento della stampante e che permette così di avere copie su carta delle operazioni effettuate.
CONTROL CODES - Permette di varia-

re gli stili di scrittura con i relativi codici che verranno poi inviati alla periferica.
SCREEN DUMP - È il comando che consente di avere una copia esatta su carta di quanto mostrato sullo schermo. Praticamente la stessa funzione sempre presente sugli Atari attivabile con la pressione di Alternate + Help.

MENU TERMINAL

In questo menu sono presenti tutti i comandi di definizione della RS232 e del modem oltre a qualche altro comando piuttosto comodo.
INDEX OF TERMINAL - Contiene l'indice di tutti i terminali, sia quelli già presenti nel programma, sia quelli preparati e salvati da noi.
LOAD/SAVE TERMINAL - Dopo avere configurato un particolare terminale è bene salvarlo per evitare di ripetere ogni volta il procedimento di preparazione. Ecco quindi la presenza di "Load" e "Save".
CONFIGURE TERMINAL - È l'opzione che permette di definire il terminale da usare. Selezionando il comando appare una finestra di dialogo che richiede diverse cose. Prima di tutto va indicato il tipo di terminale che intendiamo adottare, se questo è già presente tra quelli offerti, oppure va specificato che vogliamo crearne uno diverso. Successivamente possiamo impostare i classici parametri; la velocità, naturalmente in Baud, può essere stabilita tra 75-1200, 110, 300, 600, 1200-75, 1200, 2400, 4800, 9600.
Oltre agli altri normali dati, K-Comm 2 può variare anche il numero di colonne del testo, che può essere scelto tra 80, 64 e 40. Una cosa da notare è che il parametro Duplex qui è chiamato Echo Host (Full) e Echo Local (Half).
Alcuni elaboratori possono inviare all'inizio e al termine di ogni file un comando che avvia e ferma la registrazione. Vi sono vari modi per farlo. K-Comm 2 ne propone un paio: XON/XOFF e CTS/RTS. Se l'elaboratore a cui siete collegati possiede una di queste due possibilità, potrete selezionarla e lavorare più comodamente.
Ultima opzione presente nel Configure Terminal è il "Debug Mode", che consente di evidenziare eventuali errori di trasmissione e/o ricezione.
Tra le altre numerose possibilità offerte dal Menu Terminal troviamo, oltre agli scontati "Function keys" e "Modem Type", anche il comodissimo "Keyboard Layout", che permette di variare a piacere la disposizione dei caratteri sulla tastiera. Ad esempio potremo far sì che digitando la lettera "A" sullo schermo appaia la lettera "B" e così per tutti i tasti. Questo risulta molto comodo in un'emulazione di terminale perchè il VT52 e il VT100 utilizzano il tastierino numerico in modo diverso dal solito.

MENU OPTIONS

K-Comm 2 dispone di alcune opzioni molto utili, tra queste la più interessante è quella che permette di disporre il nostro Atari in modo "Auto-answer". Con questa funzione inserita, e ovviamente con la macchina accesa, il computer risponde automaticamente ad una eventuale chiamata da parte dell'host. "Auto-answer" protegge automaticamente i nostri file che non saranno inviati senza il nostro permesso.
Tra le altre va ricordata "Purge receive buffer" che serve a cancellare quanto contenuto nel buffer.
Interessante la possibilità di inviare all'elaboratore collegato un break al termine dei file. Il break può essere corto (Short), 275 millisecondi, o lungo (Long), 1 secondo.
Nello stesso menu vi è il comando "Disconnetted".

MENU HELP

È veramente completo, lo dimostra la lunga fila di "aiuti", e consente l'accesso a qualsiasi informazione ci sia necessaria. In questo modo molte difficoltà possono essere agilmente superate.

IMPRESSIONI D'USO

Sicuramente è più difficile da usare rispetto a FaSTcom e a Flash, ma è anche vero che offre possibilità agli altri sconosciute.
L'emulazione di terminali aiuta moltissimo l'utente che si collega spesso con grandi sistemi. Un discreto manuale e un ottimo Help aiutano comunque anche i meno esperti nella comunicazione.

UNICORN PC/INTERCOMM

Con questo programma saliamo a livelli decisamente superiori, sia di prestazioni che di prezzo. Il dischetto viene supportato da un ottimo manuale d'uso di ben 135 pagine che guida l'utilizzatore passo dopo passo verso il mondo della telematica.
PC/InterComm è piuttosto complesso, ma anche il principiante, grazie alla documentazione allegata, sarà presto in grado di destreggiarsi tra modem e pacchetti di comunicazione.
Il mouse in questo caso sparisce e al suo posto si usano i tasti direzione.
La schermata iniziale presenta il video vuoto eccezion fatta per l'ultima linea in basso, dove troviamo costantemente le indicazioni di stato.
Premendo il tasto help compare il menu principale, composto da nove varianti numerate. Per passare ad una di queste possiamo spostarci con il cursore sull'opzione desiderata oppure semplicemente digitarne il corrispondente numero.
Il programma può emulare i terminali VT52 e VT100.

COMUNICATION SPEED AND PORT SETUP

È il menu numero uno, nel quale abbiamo i consueti parametri di trasferimento. Tra i programmi fino a qui esaminati è l'unico in grado di proporre ben diciassette velocità di trasmissione che possono essere aumentate fino ad un numero indefinito grazie ad una opzione di cui parleremo in seguito. Già predisposte sono comunque 50, 75, 110, 135, 150, 200, 300, 600, 1200, 1800, 2000, 2400, 3600, 4800, 7200, 9600 e 19200 Baud.
Oltre alle selezioni di parità, di numero di bit e di bit stop, troviamo qui la possibilità di mostrare o meno gli errori di trasmissione e di lasciare o no le linee DTR e RTS.

INTERFACE SETUP

Qui possiamo selezionare l'Half o il Full Duplex, l'XON/XOFF sia in fase di spedizione che in fase di ricevimento, il modo ANSI o VT52.
Inoltre abbiamo la facoltà di far sì che il tasto backspace possa venire inviato come segnale di cancellazione (DEL) o come normale segnale di backspace (BS). Una cosa simile è possibile anche con il tasto Enter, che può essere inviato e interpretato come semplice segnale di conferma o anche come new-line.
Sempre da questo menu possiamo abbassare l'effettiva velocità di trasmissione. Se ci accorgiamo che l'host ha dei problemi a ricevere correttamente i nostri file è sufficiente correggere leggermente la velocità impostata in modo da rendere un poco più lento il flusso dei dati.

STYLE OPTIONS

Al contrario di quanto dice la parola, questo menu non consente di modificare lo stile dei caratteri, ma semplicemente di modificare alcuni parametri. In primo luogo possiamo far sì che la linea di stato venga resa invisibile oppure mostri quattro led, che visualizzano la qualità della trasmissione, al posto dell'orario. Vi è poi la funzione "Wrap", che evita di "tagliare" le parole che non stanno su una linea portandole a capo. Con l'opzione "Normal o slow scroll" possiamo modellare secondo le nostre esigenze la velocità di scrolling dello schermo, il quale può essere anche visualizzato in reverse.
Utile il suono che segnala l'avvicinarsi del fine linea e decisamente importante il comando che consente di trasformare la tastiera da UK a US a grafica e viceversa.

FILE PROTOCOL

PC/InterComm permette di utilizzare diversi protocolli di trasmissione che lo rendono molto versatile e in grado di collegarsi senza problemi a quasi tutti i calcolatori. A scelta si possono usare Mo-

dem7, Kermit, Kermit Image, PC/InterComm (solo tra due Atari con questo programma), il codice ASCII e Raw, ovvero il codice ASCII completo dei caratteri di controllo.
A volte è comodo inviare o ricevere prima o dopo ogni file una stringa di caratteri. Con InterComm è possibile grazie all'apposita funzione che chiede quale è la stringa da utilizzare.
Ultime opzioni di questo menu sono il comando "Newline" (che può essere dato con il tasto Space o con il tasto Enter) e le varie possibilità di registrazione dei file in arrivo (manuale, automatica o con la funzione XON/XOFF).

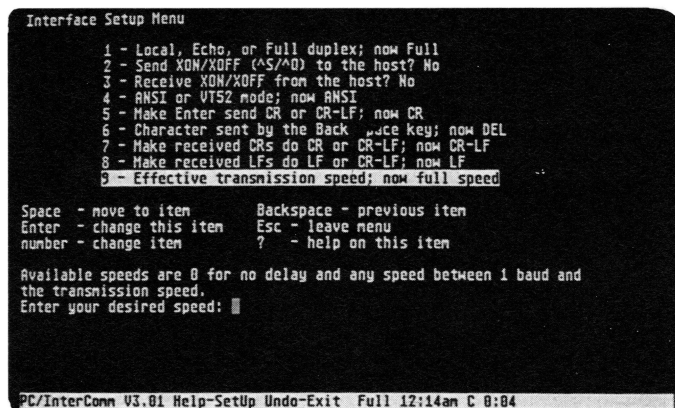
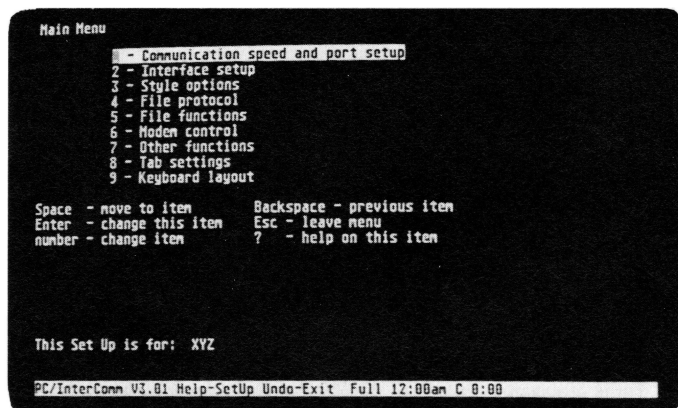
```
Main Menu

   1 - Communication speed and port setup
   2 - Interface setup
   3 - Style options
   4 - File protocol
   5 - File functions
   6 - Modem control
   7 - Other functions
   8 - Tab settings
   9 - Keyboard layout

Space  - move to item       Backspace - previous item
Enter  - change this item   Esc - leave menu
number - change item        ?   - help on this item

This Set Up is for:  XYZ

PC/InterComm V3.01 Help-SetUp Undo-Exit  Full 12:00am C 0:00
```

ctions" è quella di preparare un messaggio "di ritorno" che ci verrà inviato ogni volta che il calcolatore ospite riceverà un nostro file.

TAB SETTINGS E KEYBOARD LAYOUT

Il "Tab settings" consente di fissare dei tab in qualunque punto, anche ogni N colonne. "Layout keyboard" va utilizzata invece per ridefinire i tasti nel caso la trasformazione in tastiera inglese o statunitense non sia stata sufficiente.

IMPRESSIONI D'USO

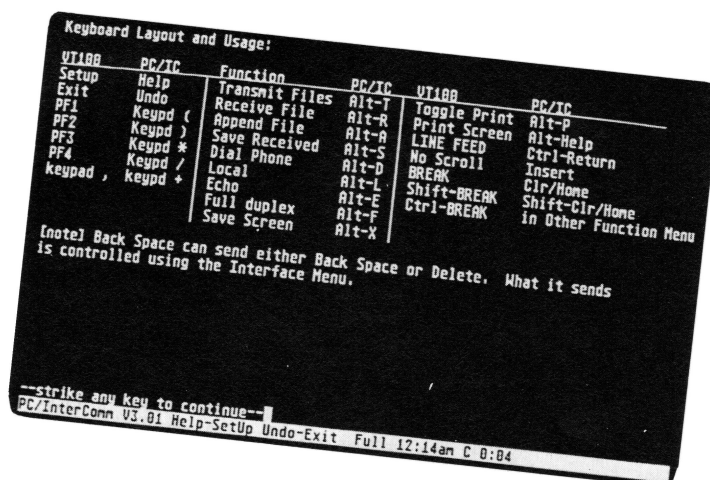
Più complesso, ma anche più completo, considerazione i prezzi.
Il più economico è Flash, della Antic che viene offerto a meno di 40 dollari. È certamente poco e il suo prezzo è giustificato non dalla bassa qualità del programma, ma dall'assenza di un emulatore.
Segue a ruota, ancora una volta tra i più economici, un programma della Kuma Computers, distribuito dalla Atari Italia. 69.000 lire, Iva compresa, è il prezzo di questo emulatore, più pacchetto di comunicazione, che non si colloca certo nella fascia bassa di prestazioni.
Con 50 sterline di prezzo d'acquisto possiamo avere FaSTcom, semplicissimo ed idcale per il neofita.

FILE FUNCTION

È il menu di controllo dei file. Serve a mandarli, a riceverli o ad aggiungerne uno nuovo a quelli già esistenti. Da notare che i file ricevuti vengono inviati ad un buffer chiamato History che può essere sempre visionato. Ovviamente la capienza di questa parte di memoria è ridotta, ma spesso, nel caso ci dimentichiamo di avviare la registrazione, può tornare utile.

MODEM CONTROL

Qui troviamo le funzioni per il collegamento con l'host computer. Oltre a "Dial the phone" e a "Phone number" abbiamo anche l'opzione per far partire automaticamente il collegamento al momento dell'accensione.

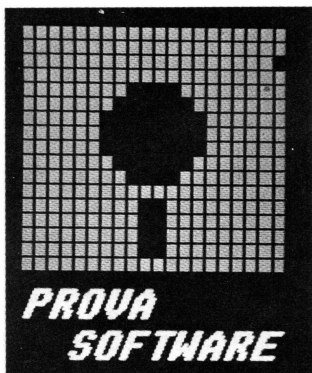
OTHER FUNCTIONS

Per nostra comodità possiamo scrivere il nome del sistema con il quale siamo collegati tramite la funzione "Computer system name" e definire i tasti funzione, sempre utili. Possiamo controllare in ogni momento cosa abbiamo assegnato ad essi con il comando "Show function keys" e, una volta definitone il contenuto, possiamo salvarli per un futuro riutilizzo.
Comoda anche la funzione di "Reset", che evita di dover spegnere il computer e quindi ricaricare il programma.
Ultima possibilità delle "Other fun-

PC/InterComm consente un'ampia flessibilità nei collegamenti, grazie all'elevato numero di protocolli che può utilizzare.
Spesso utile si dimostra l'opzione di scrolling (Slow o Normal) che consente di vedere tutti i dati ricevuti un po' più lentamente.
Necessaria, e presente, quella che permette di abbassare l'effettiva velocità di trasmissione, specialmente se la linea telefonica è disturbata.
Un programma completo con un manuale veramente notevole.

CONCLUSIONI

Un breve giudizio sulle qualità dei singoli programmi lo abbiamo già dato al termine di ciascuna prova, ora prendiamo in

PC/InterComm è il più professionale, e il più costoso, tra i pacchetti di comunicazione per Atari. Molto versatile, è corredato da un ottimo, e voluminoso, manuale in inglese che spiega tutto quanto vi è da sapere sulle comunicazioni via telefono.

Un piano più sopra invece si posiziona PC/InterComm che, con 249.000 lire, Iva compresa, si rivolge ad un'utenza più selezionata.
Come abbiamo visto ci sono programmi per tutti i gusti e per tutte le tasche. Importante è non scegliere a caso, ma pensare bene alle nostre specifiche esigenze.

PASCAL 68000

IL METACOMCO PASCAL PERMETTE DI OVVIARE A TUTTE LE LIMITAZIONI IMPOSTE DA ALTRI COMPILATORI NONCHÈ DI SFRUTTARE IN PIENO LE CAPACITÀ GRAFICHE E DI CALCOLO DELLA SERIE ST

DI ANDREA BORRONI

Per gli amanti della programmazione strutturata in PASCAL, questo prodotto è proprio quello che ci voleva. Sulla base delle esperienze passate in fatto di compilatori, la Metacomco ha sviluppato questa applicazione dedicata per soddisfare le esigenze di coloro che fanno del PASCAL il loro principale strumento per lo sviluppo di programmi. Il guadagno in velocità di risposta è qui significativamente migliorato.
È possibile poi decrementare ulteriormente, ed in maniera drastica, il tempo di compilazione e di linking se si ha la possibilità, utilizzando una famosa utility in circolazione, di installare in memoria centrale un disco virtuale, facendo poi la copia di tutti i file presenti sui due dischetti del PASCAL 68000 in questa porzione di RAM riservata.

PASCAL STANDARD

Tutti i concetti del PASCAL definiti da Wirth sono stati puntualmente rispettati in questa versione; gli statement, le parole riservate, i simboli speciali, le strutture dati, l'I/O, la questione file, la sintassi e la semantica, trovano piena corrispondenza nelle definizioni originali del linguaggio. Lo stesso dicasi per i parametri tecnici più specifici, quali l'ordine del set di caratteri, che sono pienamente conformi allo standard ISO (ASCII).
In aggiunta sono state implementate quelle estensioni che permettono una maggiore flessibilità in fase di programmazione. Ne menzioniamo qui le più significative:
- uso di identificatori di qualsiasi lunghezza (ma non distinzione tra minuscole e maiuscole);
- campo degli INTEGER pari a 2^32 (cioè un MAXINT di oltre due miliardi)
- campo dei REAL con esponente di due cifre (+/−99) in forma normalizzata
- possibilità di mettere in relazione INTEGER con REAL (risultato in REAL)
- la procedura 'NEW' per l'allocazione di variabili dinamiche è affiancata dalla procedura 'DISPOSE' che ne permette la deallocazione;
- procedure 'PACK' e 'UNPACK' per l'impaccamento e il disimpaccamento di Array e la procedura 'PAGE' per il cambio pagina;
- possibilità di dichiarazione di insieme unitamente a tipi scalari o subrange (esempio digit = SET OF 0..9);

D'altro canto ci sembra doveroso far osservare alcune limitazioni che sono state riscontrate utilizzando il compilatore:
- limitata profondità di nidificazione di procedure e funzioni (inferiore al decimo livello);
- impossibilità di usare il salto incondizionato GOTO da qualsiasi punto di un programma;

Per quanto quest'ultima nota non dovrebbe preoccupare i programmatori PASCAL, che dei GOTO non sanno che farsene, la limitazione sul livello di nidificazione di procedure e funzioni ci sembra significativa, seppur piuttosto remota. Solo scrivendo grossi programmi, infatti, si arriva a questi livelli di nesting, i quali, al limite, possono essere aggirati ritoccando l'organizzazione del programma.

L'EDITOR

Un buon compilatore di un linguaggio di programmazione non potrebbe essere definito tale se non fosse affiancato da un buon EDITOR per la scrittura dei programmi. Lo SCREEN EDITOR del PASCAL 68000 non ha molto da invidiare a quelli implementati su grossi sistemi; al solito però l'utente è costretto, per riuscire a sfruttare appieno le potenzialità del prodotto, a prendere confidenza con una carrellata di comandi definiti come combinazioni del tasto Control e di tasti alfanumerici.
Tali comandi di EDITOR possono essere inviati al sistema utilizzando due differenti modalità, Immediata o Estesa.
In modalità Immediata, come intuibile, l'effetto di un comando è subito visualizzabile; ad esempio, sono Immediate Commands l'inserimento o la cancellazione di linee, lo spostamento del cursore lungo punti del programma o il tasto Return stesso. In modalità Estesa, invece, alla quale si ha accesso in qualsiasi momento mediante il tasto Esc, è possibile programmare uno o una serie di comandi che saranno eseguiti dal sistema in maniera più o meno trasparente all'utente. Ad esempio le operazioni di salvataggio o inserimento di file, il movimento di blocchi di linee, la ricerca nel testo mediante stringhe o la sostituzione di stringhe stesse con altre, sono opzioni selezionabili in Extended Mode.
Interessante la presenza della funzione di UNDO mentre dobbiamo rilevare l'assenza di un'opzione che permetta il posizionamento all'interno di un programma

tramite flag invisibili, ad esempio flag numerici, identificanti posizioni di interesse alle quali si voglia accedere.

IL COMPILER

Una volta scritto il testo del programma, cioè il file sorgente, mediante l'editor, per produrre da questo il codice oggetto (o rilocabile) bisogna lanciare la compilazione. Normalmente al momento del lancio, un compilatore crea sul dispositivo di memoria di massa il file che dovrà contenere l'eventuale risultato esatto della compilazione.
Il compiler del PASCAL 68000 permette però di eseguire una compilazione anche senza creare alcun codice oggetto, il che può essere utile in fase di scrittura di un lungo programma, qualora si voglia di volta in volta verificare l'assenza di errori nel codice scritto fino a quel momento.
È questa una delle nove opzioni selezionabili nella linea di comando che richiama il compilatore. Questa ed altre interessanti modalità di compilazione sono selezionabili inserendo una o più di queste opzioni, che permettono, a scelta, di ottenere un listato della compilazione (opzione LIST), oppure di decidere il formato del programma oggetto a seconda del linker che si vorrà utilizzare (LINK, presente su questo package, o LINK68, fornito nel kit Atari Software Developer, ambiente CP/M-68K) o ancora, di ridirigere il file relativo ai messaggi d'errore.
Diverse scelte, quindi, che vogliono venire incontro alle diverse esigenze richieste dall'ambiente e dalle modalità di lavoro del programmatore.

IL LINKER

Anche le operazioni di linking del programma oggetto creato dalla compilazione sono relativamente semplici e veloci. Alla chiamata del linker viene richiesto di inserire la linea di comando per il controllo delle modalità di lincaggio. Anche in questo caso sono definibili diverse azioni relative alla produzione del risultato; ad esempio è possibile generare o meno una tavola dei simboli o una mappa del layout del file eseguibile.
Da notare che il software del PASCAL 68000 è stato fornito di una nutrita libreria di funzioni che permettono di utilizzare le capacità grafiche del computer. Per accedere a queste librerie, però, il comando va impartito al momento del lancio della compilazione e non in fase di linking.
Ultima osservazione: il codice eseguibile prodotto dal linker è effettivamente un file eseguibile in qualsiasi momento, senza dover fare alcuna operazione di predefinizione dell'ambiente di lavoro.

CONCLUSIONE

Un uso accurato e continuo di PASCAL 68000 permetterà di scrivere programmi di elevato livello professionale e di alta funzionalità. Non c'è quindi molto da aggiungere se non il consiglio di familiarizzare con tutti i comandi prima di dare inizio alla stesura di un lungo progamma, così da trovarsi in condizioni di sveltire le operazioni di editing, di compilazione e di linking.

Nome: MCC Pascal 68000
Produttore: Metacomco Ltd. (GB)
Distributore: Hard & Soft, Terni
Tipo: linguaggio di programmazione
Configurazione: Atari 520 ST, 1 drive
Prezzo:

USO E ABUSO DI UN FLOPPY DISK

DI ROBERTO CAZZARO

LA SECONDA PUNTATA DEL RAPIDO CORSO IN C SULLA STRUTTURA E IL FUNZIONAMENTO DEL FLOPPY DISK ATARI ST.

Sapevo che prima o poi sarei stato costretto a porgere le mie scuse ai lettori a causa di un errore in un programma, ma speravo che accadesse il più tardi possibile. Invece eccomi qui, alla seconda puntata, a cospargermi il capo di cenere a causa di un banale errore.
Mi spiego meglio: come faccio d'abitudine, ho distribuito fra vari amici il mio programma perché ne verificassero il perfetto funzionamento e, avendo ricevuto risposta positiva, ho fatto avere il programma al direttore della rivista per la pubblicazione; dopo qualche giorno uno dei miei amici mi viene a dire che a lui il mio programma sembra un schifezza perché non si comporta come descritto nell'articolo; mi metto subito al lavoro con lui, ma a casa mia tutto è ok; dopo alcune ore perse nel verificare le ipotesi più assurde (influenza della posizione astrale di un computer rispetto all'altro, oroscopo dell'operatore ecc.), provo a chiedere: "Ma tu che Basic usi?" – "Il Gfa Basic come mi hai detto" – "Si, ma quale versione?" – "La versione 1.0" - Boink!
E così si spiega tutto: infatti la versione 1.0, oltre ad avere molti "bug" più o meno nascosti, manca di alcune istruzioni e altre si comportano in maniera diversa; nel mio caso le differenze riguardano le costanti True e False (non definite nella prima versione) e la funzione Inp(n) che restituisce solo i codici Ascii (a proposito, sapete come si pronuncia Ascii? con la 'c' dura, visto che gli anglosassoni non hanno la 'c' dolce, quindi qualcosa come 'askii') e non il valore dei tasti tipo frecce e simili.
Conoscere bene il problema è come averne già la soluzione e i possessori della versione 1.0 potranno digitare le modifiche della tabella 1; in ogni caso è possibile richiedere la versione 2.0, e io lo consiglio, spedendo il disco originale alla Gfa e richiedendo l'ultima release; questo se si è in possesso della versione originale, altrimenti....
Il modo migliore per controllare quale versione si possiede, consiste nello scrivere, a memoria libera e in 'direct mode', la seguente linea di programma:
Print True

Tabella 1

Correzioni al programma della puntata precedente, per i possessori del Gfa basic versione 1.0
Inserire verso l'inizio del programma, dopo
Logical=0
e prima di MAIN, le seguenti linee
True-1
False=0
e sostituire la procedura WAIT
con la seguente:

```
Procedure Wait
    Repeat
        R=Gemdos(7)
        Rh=(R Div 65536) Mod 256
        R=R Mod 256
        If Rh>=71 And Rh<=90 Then
            R=Rh+128
        Endif
    Until R=27 Or R=200 Or R=208 Or
    R=203 Or R=205 Or R=199 Or
    (R>=48 And R<=57)
Return
```

La funzione Gemdos(7) restituisce una word che contiene lo Scan-code del tasto nel byte alto e il codice Ascii nel byte basso.
Con queste modifiche il programma funzionerà perfettamente.

La risposta sarà 0 per le versioni precedenti la 2.0 e −1 per le versioni seguenti la 2.0.
Detto questo iniziamo a trattare l'argomento della puntata.

UN PO' DI RIASSUNTO

Cominciamo col ripetere, brevemente, i punti principali esposti nel precedente articolo:
– Il disco, mediante formattazione, viene diviso in settori e tracce, generalmente 80 tracce e 9 settori.
– Per comodità si trattano solo i settori logici, 720 o 1440 a seconda che il disco sia a singola o a doppia faccia, e i cluster (= insieme di due settori logici consecutivi).
– Ogni settore logico contiene, oltre ai segnali di sincronismo, 512 byte di dati.
Tutto ciò, comunque, non spiega come sia possibile capire quale tipo di disco si sta usando, né come si acceda ai vari file presenti su di esso. Scopo di questo secondo articolo è quello di spiegare l'organizzazione del disco, così come viene visto dal sistema operativo.
Vi sarà capitato di leggere che il 'Boot sector' dell'Atari è compatibile con quello dell'IBM. Bene, ma cosa significa tutto questo?

IL BOOT SECTOR

Ogni sistema operativo che si rispetti, riserva una parte del disco alle informazioni relative al tipo di formattazione usata, alle caratteristiche fisiche del disco e ad altre informazioni accessorie; di solito questi dati sono memorizzati in un settore, il cosiddetto Boot sector (cioè settore che viene letto all'accensione e inizializ-

```
' -----------------------------------------------------------------+
'        Programma per la lettura e la stampa del Boot sector     !
'                                                                  !
'              Prima versione:    19/3/1987                        !
'                                                                  !
'              Autore:      Roberto "Headache" Cazzaro             !
'                                                                  !
' -----------------------------------------------------------------+
'
' Funzione per la lettura di un settore logico (da 0 a 720)
' Non viene fatto nessun controllo se il parametro passato e` compreso nei
' limiti quindi si ponga attenzione a non inserire Logical maggiore di 730!
'
Deffn Read_sect(Logical)=Xbios(8,Buf_high,Buf_low,0,0,Devno,Logical Mod 9+1,Logi
cal Div 9,Sideno,1)
'
' Funzione per leggere una word in memoria con la convenzione Intel
'
Deffn Ipeek(Ind)=Peek(Ind)+256*Peek(Ind+1)
'
' Dimensionamento Buffer e inizializzazione variabili
'
Dim A%(256)
Buffer=Varptr(A%(0))
Buf_high=Buffer Div 65536
Buf_low=Buffer Mod 65536
Sideno=0
Devno=0
'
' MAIN
'
Cls
Gosub Pr_info
Repeat                        ! attende la pressione di un tasto e torna
  '                           ! all'editor
Until Inkey$<>""
Edit
'
' Procedura per la stampa del Boot sector
'
' Stampa tutte le informazioni contenute nel Boot sector
' Per maggiori informazioni sul significato si veda la tabella 2
'
Procedure Pr_info
  Ret=Fn Read_sect(0)     ! legge il settore logico 0
  Print "Filler = ";      ! stampa il nome del costruttore, non usato
  For I=2 To 7
    Print Chr$(Peek(I+Buffer));
  Next I
  Print
  Print "Serial number = ";Dpeek(Buffer+8)*256+Peek(Buffer+10)
  Print "Bytes/sect = ";Fn Ipeek(Buffer+11)
  Print "Sectors/cluster = ";Peek(Buffer+13)
  Print "Reserved sectors = ";Fn Ipeek(Buffer+14)
  Print "Number of fats = ";Peek(Buffer+16)
  Print "Directory entries = ";Fn Ipeek(Buffer+17)
  Print "Sectors on media = ";Fn Ipeek(Buffer+19)
  Print "Media type = ";Peek(Buffer+21)
  Print "Sectors per FAT ";Fn Ipeek(Buffer+22)
  Print "Sectors per track = ";Fn Ipeek(Buffer+24)
  Print "Number of sides = ";Fn Ipeek(Buffer+26)
  Print "Hidden sectors = ";Fn Ipeek(Buffer+28)
Return
```

Tabella 2

Il Boot sector contiene:
– un numero seriale che identifica univocamente (si spera) il disco in uso e che viene usato per verificare il cambio disco
– i parametri del bios (dati del disco)
– ilcodice, opzionale, di bootstrap, cioè quel codice che viene usato per eseguire automaticamente un programma all'accensione.
Gli ultimi due byte servono affinché il checksum (= somma per verificare l'esattezza dei dati contenuti nel settore) sia uguale a $1234. Per maggiori informazioni si veda "A Hitchhiker's guide to the Bios".

Contenuto	locazione	commento
Bra.s nnnn	$00	Salta al boot code, più avanti nel settore
Filler	$02	Riservato al nome del costruttore, non usato
Serial	$08	Numero seriale del disco
Bps	$0b	Numero di settori per settore
Spc	$0d	Numero di settori per cluster
Res	$0e	Numero settori riservati al Dos
Nfats	$10	Numero di Fat presenti
Ndirs	$11	Numero massimo di file nella directory
Nsects	$13	Numero di settori presenti sul disco
Media	$15	Descrittore del disco, non usato
Spf	$16	Numero di settori riservati alla Fat
Spt	$18	Numero di settori per traccia
Nsides	$1a	Numero di facce del disco
Nhid	$1c	Numero di settori "nascosti"
Boot	$1e	Inizio del boot code
Check	$1fe	Bytes di checksum

zazione del sistema). Il boot sector permette di usare le più disparate formattazioni e tipo di dischi (3.5″ o 5.25″) senza grossi problemi di compatibilità; ad esempio l'IBM utilizzava dischi a 40 tracce, 8 settori e doppia faccia (=320 Kbyte) nella versione 1.0 del PC-DOS, mentre ora utilizza 40 tracce, 9 settori a doppia faccia (=360 Kbyte), senza modifiche al drive o ai programmi di gestione.
Il Boot sector può, eventualmente, contenere un piccolo programma in linguaggio macchina, detto "loader" o "Boot code" che, se presente, esegue automaticamente l'inizializzazione del sistema e "prende le redini", di solito richiamando un programma più complesso residente sul disco: era ciò che accadeva quando il nostro Atari, non avendo il sistema operativo in ROM, era costretto a caricarselo ogni volta da disco. Se provate a confrontare il boot sector di un disco con il vecchio sistema operativo con un qualunque altro disco, noterete una profonda differenza.
Il Boot sector è il settore logico 0, cioè il primo ad essere letto e corrisponde sempre al settore 1, traccia 0, lato 0, sia per dischi a singola che a doppia faccia.
Per visualizzare la struttura del boot sector si veda la tabella 2.
La compatibilità IBM si riferisce al fatto che le informazioni relative al tipo di disco, numero settori ecc. sono disposte come nel Boot sector di un disco IBM: ciò significa che se si inserisce un disco a 3.5″ di un IBM Ventiquattrore (il modello trasportabile), è possibile leggere le informazioni e i file ivi contenuti, naturalmente solo il file testo in Ascii. Se, come me, attaccate un drive da 5.25″ (e prima o poi spiegherò come), potrete leggere anche i dischetti dell'IBM normale.
Dopo questa breve parentesi, siete pronti per inserire il programma numero 1, relativo alla procedura di stampa informazioni del boot sector. A proposito, in questa puntata non vi darò un programma completo, come la volta precedente, ma un insieme di procedure e funzioni, che vi aiuteranno a capire meglio la struttura del disco; la prossima puntata sarà interamente dedicata ad un programma monitor per i dischi, che vi permetterà di leggere e modificare i vostri, e non solo vostri, file.
Un problema che si incontra nel leggere il Boot sector è che, per mantenere la compatibilità IBM (e vi assicuro che ne vale la pena), le word (= parole di 2 byte) seguono il formato Intel, cioè il byte più importante viene per secondo e non, come in casa Motorola, per primo: in poche parole i due bytes sono in ordine inverso; per ovviare all'inconveniente ho scritto la funzione I-peek (address) che si incarica di sistemare le cose (I-peek stà per Intel-peek e si usa come un Dpeek - address - normale).
Del programma resta poco da dire, visto che è semplice e molto commentato, pronto per essere eseguito.

```
' ---------------------------------------------------------------------------+
'         Programma per la lettura e la stampa della Directory               !
'                                                                            !
'              Prima versione:   19/3/1987                                   !
'                                                                            !
'              Autore:      Roberto "Headache" Cazzaro                       !
'                                                                            !
' ---------------------------------------------------------------------------+
'
' Nuova funzione per la lettura di un settore logico (da 0 a 720 o a 1440)
' Non viene fatto nessun controllo se il parametro passato e` compreso nei
' limiti quindi si ponga attenzione a non inserire Logical maggiore di 730
' se il disco e` a singola faccia e 1450 se il disco e` a doppia faccia
'
Deffn Read_sect(Logical)=Xbios(8,L:Buf,L:0,Devno,Logical Mod Spt+1,Logical Div S
ector,(Logical Mod Sector) Div Spt,Count)
'
' Funzione per leggere una word in memoria con la convenzione Intel
'
Deffn Ipeek(Ind)=Peek(Ind)+256*Peek(Ind+1)
'
' Dimensionamento Buffer e inizializzazione variabili
'
Dim A%(256)
Buf=Varptr(A%(0))
Spt=9
Sector=9
Devno=0
Count=1
'
' MAIN
'
Cls
Gosub Get_info
Gosub Pr_dir
Edit
'
' Procedura per la stampa della directory.
'
' Stampa il nome del file, se e` una directory o un file normale,
' il settore logico di partenza e la lunghezza del file.
'
' N.B. stampa anche i file cancellati, facendoli precedere da una
' sigma minuscola.
'
Procedure Pr_dir
  Logic=Disp
  Flag=0
  Ret=Fn Read_sect(Logic)
  Counter=Buf
  Repeat
    If Ret=0 Then
      Print
      Print "Nome            tipo    sl     lunghezza"
      Print
      Repeat
        Gosub Get_dosname(Counter)
        Print Dsname$;
        Add Counter,11
        If Peek(Counter)=16 Then
          Print " [DIR] ";
        Else
          Print " [    ] ";


        Endif
        Add Counter,15
        Print Right$("    "+Str$(Fn Ipeek(Counter)*2+14),4);
        Add Counter,2
        Print Right$("      "+Str$(Fn Ipeek(Counter)+65536*Fn Ipeek(Counter+2)),1
0)
        Add Counter,4
      Until Peek(Counter)=0 Or Counter=Buf+512
    Else
      Gosub Errore
      Flag=1
    Endif
    Inc Logic
    Ret=Fn Read_sect(Logic)
    Counter=Buf
    Gosub Wait
  Until Peek(Counter)=0 Or Flag=1
Return
'
' Procedura per la lettura di un nome secondo le convenzioni del Dos
'
' Si noti che la directory e i nomi seguono il formato IBM
'
Procedure Get_dosname(Ind)
  Dsname$=""
  For I=Ind To Ind+7
    Dsname$=Dsname$+Chr$(Peek(I))
  Next I
  Dsname$=Dsname$+"."
  For I=Ind+8 To Ind+10
    Dsname$=Dsname$+Chr$(Peek(I))
  Next I
Return
'
' Procedura che attende la pressione di un tasto, funziona sia con
' la versione 1.0 che con 2.0.
'
```

```
Procedure Wait
  Local R
  R=Inp(2)
Return
'
' Procedura per la lettura dei parametri del disco e della funzione Read_sect
'
' si veda l'articolo per spiegazioni.
'
Procedure Get_info
  Ret=Fn Read_sect(0)
  If Ret=0
    Sect_no=Fn Ipeek(Buf+19)
    Sides=Fn Ipeek(Buf+26)
    Spt=Fn Ipeek(Buf+24)
    Fat_lenght=Fn Ipeek(Buf+22)
    Disp=Peek(Buf+16)*Fn Ipeek(Buf+22)+1
    Sector=Spt*Sides
  Else
    Gosub Errore
  Endif
Return
'
' Procedura di stampa errore e ritorno a editor
'
Procedure Errore
'
Mesg$="Errore numero "+Str$(Ret)+".   ! !Ritorno all'editor"
Alert 3,Mesg$,1," EDITOR ",Dummy
Edit
Return
```

UNA NUOVA READ_SECT

Il programma della puntata precedente, nonostante avessimo parlato di dischi a singola e a doppia faccia, poteva leggere solo una faccia del disco, la superiore o inferiore, a seconda del valore della variabile Sideno. A partire da questa puntata, tratterò correttamente i dischi, qualunque tipo siano, leggendo le informazioni relative nel Boot sector e scrivendo una nuova funzione Read_sect, in modo da tener conto di tutte le possibili configurazioni. Cuore di tutto è la procedura Get_info, che si incarica di leggere il Boot sector e di estrarne i valori caratteristici, assegnandoli alle variabili Spt (= settori per traccia), Disp (= displacement, settore logico di inizio della directory), Sect_no, numero di settori sul disco e Sector, che contiene il numero di settori per traccia moltiplicato per il numero di facce, visto che il sistema operativo gestisce la seconda faccia come una estensione della prima: dove finisce una traccia, lì comincia la stessa traccia, ma sull'altro lato, per minimizzare gli spostamenti della testina, punto debole del disco. La funzione Read_sect legge il disco, convertendo il settore logico in settore, traccia e faccia fisica del disco; con un minimo di prove dovreste poter risalire alle formule usate per convertire Logical. Leggendo più attentamente i manuali, in tedesco, mi sono accorto che, a differenza di quanto scritto nella puntata precedente, è possibile passare una long word ad una funzione, semplicemente facendo precedere la variabile da L:; così si spiega quel misterioso L:0, che sta per 4 byte (= long word) a zero. Sempre a proposito dei manuali, consiglio la frequente rilettura di tutti quelli in vostro possesso. Ho scoperto che molte persone non conoscono l'uso del tasto Esc da Desktop: premendo il tasto Esc, quando c'è una finestra attiva, si ottiene l'aggiornamento del contenuto della finestra, mediante rilettura del disco nel drive corrente, molto più comodo che non chiudere e riaprire la finestre ogni volta che si cambia un disco.

Tabella 3		
Struttura della directory		
Descrizione	**lunghezza**	**commenti**
Nome del file	8 bytes	In caratteri Ascii maiuscoli
Estensione	3 bytes	In caratteri Ascii maiuscoli
Attributo	1 byte	Vedi oltre
Riservato	10bytes	Riservto per usi futuri (a 0)
Ora e data	4 bytes	Data dell'ultima modifica
Cluster	2 bytes	Numero del primo cluster
Lunghezza	4 bytes	Lunghezza del file
Descrizione degli attributi		
valore numerico	**significato**	
00		Read-Write / visibile / normale
01		Ready only / visibile / normale
02		Read-Write / nascosto / normale
03		Read only / nascosto / normale
04		Read-Write / visibile / di sistema
05		Read only / visibile / di sistema
06		Read-Write / nascosto / di sistema
07		Read only / nascosto / di sistema
08		Volume label
16		Folder (subdirectory)

LA DIRECTORY

Passo ora a descrivere uno dei settori più importanti, la directory, che si incarica di contenere i dati relativi alla posizione di ogni file all'interno del disco. Ho parlato di settore, anche se in realtà la directory del nostro disco si estende per 7 settori, dal settore logico 11 al settore logico 17 compresi. Siccome però in alcuni dischi non standard è possibile che sia in altra posizione, il settore di partenza si calcola moltiplicando il numero delle Fat (File Allocation Table, tabella di posizione dei file) per la loro lunghezza e aggiungendo uno al risultato; nel programma io assegno questo valore alla variabile Disp. La directory contiene, per ogni file, il nome, il tipo (se file, subdirectory, file protetto ecc.), la data di creazione, il cluster di partenza e la lunghezza come viene descritto in tabella 3. Come al solito, ho scritto un breve programma per la lettura delle informazioni nella directory e ne ho fornito il listato (listato numero 2); dopo aver lanciato il programma appaiono sullo schermo tutti i file presenti su disco (20 alla volta, per vederne altri venti si prema un tasto qualunque), seguiti dal tipo di file (solo se subdirectory appare la scritta DIR), dal settore logico di partenza (ricavato con una formula empirica, a partire dal numero di cluster) e dalla lunghezza del file. Si notino due cose: primo, anche la directory segue lo standard Intel per i numeri e, secondo, i file cancellati non vengono fisicamente rimossi, ma ne viene posto il primo carattere a $E5, 229 in decimale, a segnalare che quella "entry" è libera (nell'Atari i settori, alla formattazione, vengono posti a 229 e non a 0, per avere un migliore controllo sui settori difettosi). Ho volutamente trascurato di interpretare la data e i tipi di file in modo da non appesantire troppo un programma che serve solo come demo della struttura della directory, anche se ho dato descrizione del tipo di file, il cosiddetto attributo, nella tabella 3.

Buon lavoro, dunque, e non esitate a favi vivi se avete qualche problema con i listati.

BOFFIN: SCRIVERE SENZA LIMITI

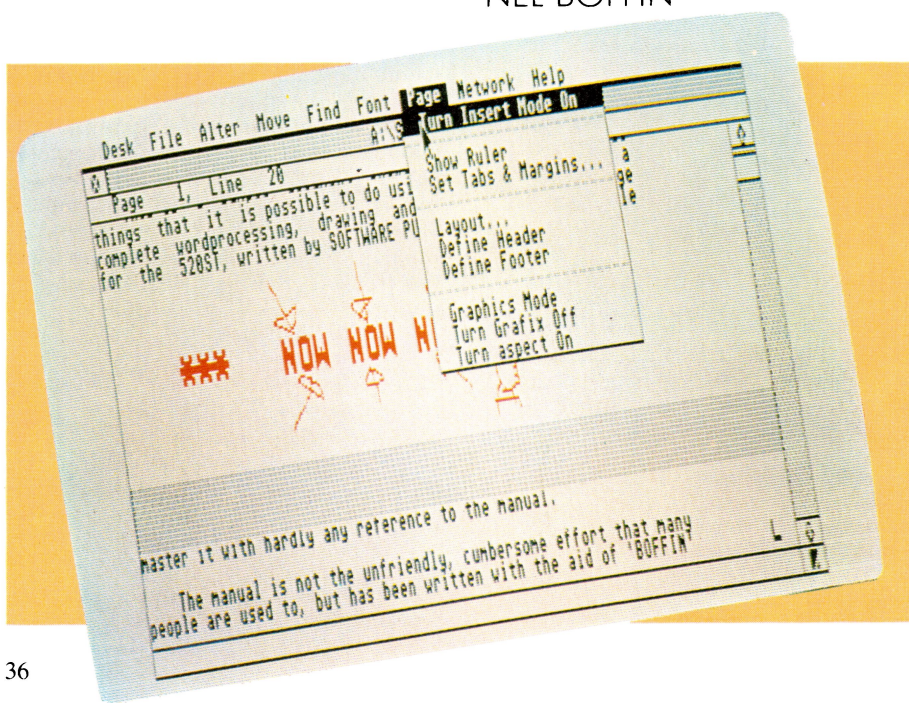
DI MAURO SCAIONI

BOFFIN È UN WORD PROCESSOR FRA I PIÙ EVOLUTI ADATTO ALL'ATARI ST. È BASATO SUI CLASSICI MENÙ PULL DOWN DEL GEM, CON IL QUALE SI INTEGRA PERFETTAMENTE COME FINESTRA DI FUNZIONAMENTO. CONSENTE NATURALMENTE L'INSERIMENTO DI GRAFICI ED IMMAGINI, IL CUI DISEGNO È POSSIBILE GRAZIE AD UN SOTTOPROGRAMMA INTEGRATO NEL BOFFIN

Boffin è un nome piuttosto frivolo per uno dei più evoluti programmi di word processor disponibili per i sistemi Atari della serie ST. Il programma è prodotto dall'americana Software Punch e commercializzato in Italia dalla Hard & Soft di Terni (tel. 0744-46658). Le sue caratteristiche gli consentono di integrarsi alla perfezione con il sistema operativo GEM della macchina. Il Boffin gira infatti in una *window* che può essere ingrandita e rimpicciolita a piacimento così come può essere sovrapposta ad altre finestre in cui girano differenti programmi o un'altra task del Boffin in modo da poter elaborare più testi alla volta. Del sistema GEM il Boffin mantiene tutte le caratteristiche operative, dall'utilizzo del mouse all'impiego dei menu *pull-down* per la programmazione delle diverse funzioni. Un'ulteriore feature del programma è quella di poter inserire nel testo grafici e disegni. Questa capacità, comune a molti word processor dell'ultima generazione, è però particolarmente curata nel Boffin, che prevede persino una finestra grafica per disegnare direttamente le illustrazioni o per generare automaticamente diagrammi a torta o a barre.

MODALITÀ TEXT

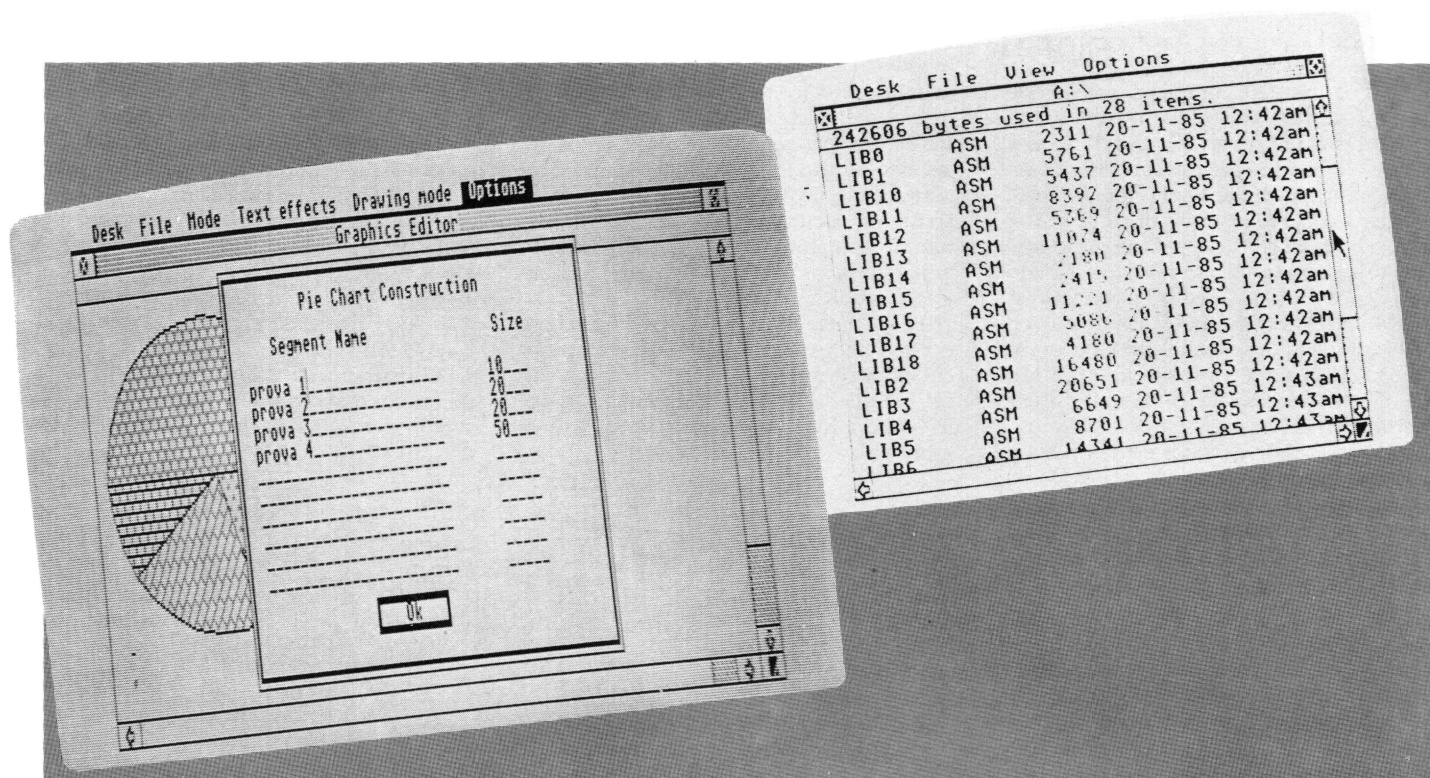
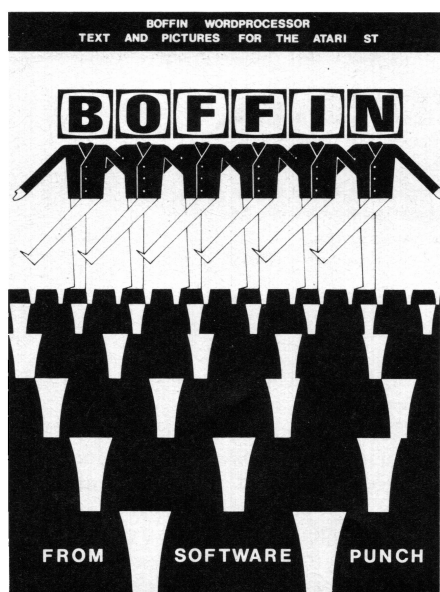
Uno dei pregi maggiori del Boffin è sicuramente la facilità d'uso. Una volta lanciato è tutto fatto: si può tranquillamente iniziare a scrivere. Sullo schermo sono normalmente presenti due cursori: quello del testo, a forma di L se si è in modalità *insert* o a forma di I se ci si trova in *overwrite*, e quello del mouse. Quest'ultimo consente di *puntare* i menù pull down e di avere accesso quindi alle varie funzioni del programma. I menù disponibili sono 9, corrispondenti ad altrettante grandi famiglie di comandi:

Il segreto della semplicità di utilizzo di questo software sta nella presenza dei menù pull-down. È come avere sempre a disposizione un elenco completo dei comandi utilizzabili.

DESK: dà accesso a quelle funzioni del GEM che siano state precedentemente attivate.
FILE: permette di compiere tutte le operazioni sui file: caricamento, concatenamento, cancellazione... In più dà accesso anche all'inserimento nel testo di finestre grafiche precedentemente memorizzate su disco ed alla gestione di tutte le funzioni di stampa. Sono infatti disponibili due differenti tipi di stampa, quella veloce (Quick Print), dove non vengono riportati gli effetti speciali sul testo o eventuali immagini inserite, e quella normale (Full Print), dove vengono abilitati tutti gli effetti speciali e le immagini grafiche introdotte nel testo. Sempre da questo menu è possibile definire il tipo di stampante che si intende utilizzare. Si possono sia richiamare le caratteristiche di alcuni modelli (già presenti sul dischetto), sia configurarne di completamente nuovi inserendone le caratteristiche in un'apposita maschera sullo schermo.
ALTER: da questo menu è possibile aggiungere o cancellare blocchi o righe di testo in qualsiasi punto dell'elaborato.
MOVE: come suggerisce il nome stesso, questo menu predispone tutte le funzioni di movimento dei blocchi di testo: copia, taglio, centramento, riassestamento...
FIND: consente di ricercare e/o modificare in modo automatico qualsiasi gruppo di caratteri nel testo.
FONT: da questo menu è possibile introdurre effetti speciali nel testo: italico, ribattuto, sottolineato, allargato... Il cambiamento vero e proprio del tipo di carattere deve invece avvenire caricando il *file-font* dal disco.
PAGE: abbassando il menu *Page* si ha accesso a tutta una serie di comandi concernenti il layout della pagina dattiloscritta più alcune funzioni accessorie (come l'inserimento di un cursore graduato nella fascia alta della pagina ed il passaggio al modo grafico).
NETWORK: consente di scambiare testi o parti di testo tra due o più computer collegati in rete.
HELP: dà accesso a schermi di aiuto per le varie situazioni operative del programma.

Il processore grafico consente non solo di realizzare le illustrazioni desiderate tramite il mouse, ma anche di importare immagini da altri programmi.

Una delle caratteristiche del Boffin è quella di mostrare sullo schermo in modo completamente fedele quello che poi verrà stampato sul foglio cartaceo.

MODALITÀ GRAPHICS

Per passare al funzionamento in modalità grafica, è necessario selezionare l'apposita opzione del menu PAGE, che fa apparire una finestra grafica che occupa i 2/3 della parte bassa dello schermo. Nel contempo i menu *pull--down* vengono sostituiti da nuove versioni contenenti i comandi necessari alla grafica:
DESK: ha il medesimo significato che nella modalità testo.
FILE: in questo caso vengono consentite le operazioni di caricamento e salvataggio delle immagini grafiche. Esse possono avere quattro differenti formati per rendere il Boffin compatibile con i più diffusi programmi di grafica computerizzata: si possono caricare cioè immagini con file Doodle, Degas e Metafiles nonché, naturalmente, Boffin.
MODE: il menu Mode permette di passare da comandi di disegno a quelli di testo (sempre nell'ambito della tavola grafica), con la possibilità inoltre di cancellare, spostare o copiare porzioni del disegno stesso.
TEXT EFFECTS: la possibilità di scrivere testi nella finestra grafica è notevolmente sviluppata proprio per la capacità del Boffin di sintetizzare un grande numero di caratteri ed alterazioni grafiche degli stessi.
DRAWING MODE: da questo menu si scelgono le entità da disegnare mediante il mouse: linee, cerchi, ellissi, rettangoli.... Quello che però è fuori dal comune, è la possibilità di disegnare in modo automatico diagrammi a torta o a barre semplicemente scrivendo le differenti variabili in maschere che compaiono sullo schermo.
OPTIONS: questo menù consente la scelta di alcuni parametri di funzionamento: si possono variare i font (della matrice dei caratteri) della funzione text della lavagna da disegno, ed è anche possibile determinare lo spessore e il tipo di linea del disegno o il pattern impiegato dalla funzione di riempimento di area del progamma.
La potenza del processore grafico del Boffin, sta soprattutto nella possibilità di operare anche con immagini *prodotte* da altri programmi. Nel testo esplicativo fornito con il dischetto del Boffin sono inseriti diversi esempi di grafica tra cui una foto passata al digitalizzatore: la qualità è eccellente e la possibilità di sfruttare appieno le avanzate possibilità grafiche del sistema Atari ST rendono questo programma tra i più adatti per il *Personal Publishing*. Basta infatti avere

Da sottolineare la possibilità di generazione automatica di diagrammi a torta o a barre.

Il dischetto del Boffin è accompagnato da un manuale d'uso redatto in lingua inglese. L'utilizzo del manuale è comunque limitato alle fasi di approccio al programma, dato che la facilità d'uso del Boffin e la disponibilità di Help in linea ne escludono la necessità di consultazione durante l'impiego del programma.

una stampante adeguata (per esempio un modello a laser) per poter realizzare, dalla fase di impaginazione a quella di preparazione delle patinate, una rivista o ancor più facilmente, un libro. Manca, a nostro parere, la possibilità di agganciarsi ad un qualsivoglia database in modo da poter eseguire mailing personalizzati. Tale mancanza è però comune a quasi tutti i programmi di word processing della classe del Boffin.

IMPRESSIONI D'USO

Quello che più colpisce chi si avvicina per la prima volta al Boffin è la sua semplicità d'uso. Dopo pochissimi minuti che lo si adopera pare di averci sempre lavorato. Questo è dovuto all'intuitivtà del sitema mouse + menu pull-down che non richiedono lo sforzo mnemonico di mantenere in mente qualsivoglia comando: sono sempre tutti pronti e disponibili, basta esplorarli col mouse!
La possibilità di sbizzarrirsi con differenti caratteri e dimensioni degli stessi può creare una iniziale confusione nell'utilizzatore non abituato a questo genere di funzione.
Passata questa fase di training però la *flessibilità tipografica* del Boffin non tarderà a farsi apprezzare. Molto intuitivo risulta anche l'utilizzo del processore graficò che permette di realizzare con la massima velocità le illustrazioni per i propri testi.
Se a questo aggiungiamo che quello che appare sullo schermo è ciò che realmente verrà stampato sul nostro foglio di carta, si capisce come Boffin sia un ottimo programma di videoscrittura.
Unendo queste particolarità si ottiene una versatilità di lavoro senza eguali.
L'unico appunto che forse si può rivolgere a Boffin è che per cambiare lo stile di scrittura è necessario digitare tutto il documento in caratteri normali e solo successivamente, selezionando la parte di testo da trasformare, è possibile avere i caratteri italici, sottolineati, in grassetto e così via. Non è ovviamente un grosso difetto, ma questa operazione può costare un po' di tempo.
Molto semplice risulta invece l'operazione di spostamento dei margini, che si effettua tramite mouse "trascinando" l'indicatore nella posizione voluta. Realizzare un'immagine da inserire nel nostro documento è oltremodo semplice e veloce grazie alle apposite funzioni implementate nel WP.

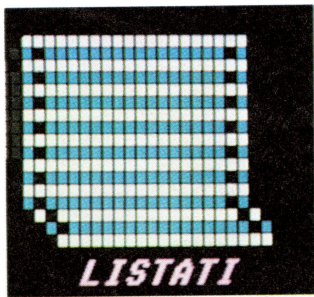
CONCLUSIONI

Il Boffin risulta essere un programma di stesura testi molto potente ed articolato. Nonostante la notevole flessibilità, l'apprendimento del suo utilizzo è rapido e *indolore*. I particolari che più si fanno apprezzare sono la simulazione completa (compreso il tipo di carattere e di corpo) del testo stampato sul video del computer e la possibilità di creare e gestire direttamente dal programma illustrazioni e grafici di corredo. A proposito dei grafici rimarchiamo l'importantissima capacità del Boffin di generare in modo automatico diagrammi a torta ed a barre.
Prima di chiudere va fatta un'ultima considerazione sul prezzo, che, viste le prestazioni del prodotto, si potrebbe credere elevato. Invece è una lieta sorpresa: con 74.000 lire (Iva esclusa) possiamo iniziare a lavorare con il nostro Boffin!

QUESTIONARIO

In questa pagina trovate un tagliando per poter partecipare attivamente alla vita della vostra rivista preferita: potete ritagliarlo e poi spedirlo a La Rivista di Atari - By Byte - Corso di P.ta Romana 1 - 20122 Milano

1. Ti piace la rivista di Atari? ☐ molto ☐ abbastanza ☐ poco
2. A quali rubriche daresti più spazio ______
3. Che cosa non ti piace? ______
4. Vorresti nuove rubriche? Se sì, quali? ______
5. Ti piacerebbe che la rivista di Atari fosse ☐ quindicinale ☐ mensile ☐ bimestrale
6. Compri altre riviste del settore? Quali? ______
7. Quanti programmi per Atari hai comprato negli ultimi 6 mesi? ______
   videogiochi ______
   altri programmi ______
8. Su quale supporto? ☐ cassetta ☐ disco
9. Quanti programmi possiedi? ______
10. Quanti di questi sono originali? ______
11. Possiedi un computer? Se sì, quale? ______
12. Usi un computer non tuo? Se sì, quale? ______
13. Quale configurazione possiedi? ☐ registratore ☐ floppy ☐ stampante ☐ monitor b/n ☐ monitor colori ☐ TV color ☐ TV b/n ☐ plotter ☐ altro
14. Quale joystick possiedi? ______
15. Possiedi un modem o un accoppiatore acustico? ☐ modem ☐ accoppiatore acustico
16. Ti colleghi con qualche banca dati? Se sì quale? ______

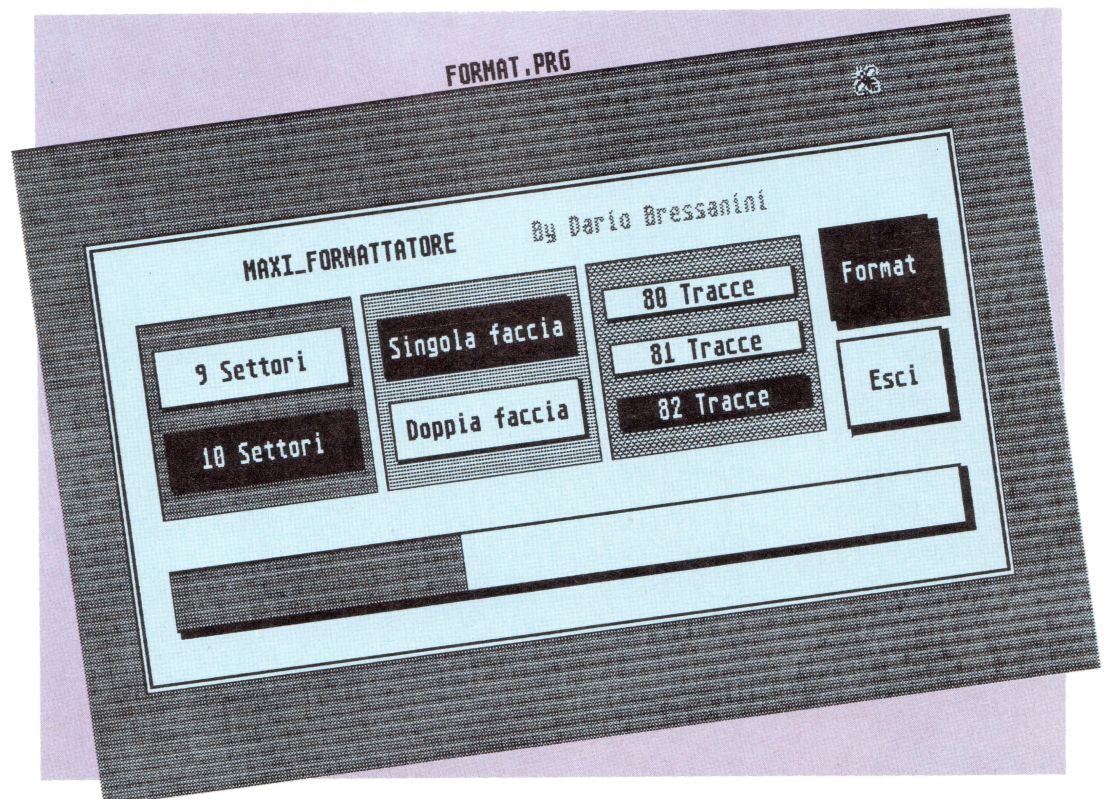
NOME ______ COGNOME ______
Età ______ Professione ______
Via ______ N° ______ Città ______

I SEGRETI DEL GEM

TUTTO IL GEM MINUTO PER MINUTO ATTRAVERSO I PROGRAMMI IN C CHE SFRUTTANO A FONDO LE ROUTINE DI SISTEMA

DI DARIO BRESSANINI

Forse sarà capitato anche a voi di pensare "Mi piacerebbe poter sfruttare nei miei programmi i menù a discesa, le finestre e tutte le altre cosine che l'Atari ST e il GEM mi mettono a disposizione, ma chissà come diavolo si fà!!". Se siete tra questi, allora siete proprio fortunati: questo articolo e quelli che seguiranno sono ciò che fa per voi.
L'Atari è una splendida macchina, veloce (più del PC IBM), verstile, potente ed è un vero peccato non saperla sfruttare al meglio.
Per poter utilizzare i menu, le icone, le finestre e tutto il resto, bisogna imparare a programmare in GEM, cioè ad utilizzare le routines contenute nelle ROM del computer. Niente paura, non è necessario utilizzare il linguaggio macchina, si può benissimo usare un linguaggio ad alto livello: C, Pascal, Basic; beh, a dire il vero il Basic non è proprio il linguaggio adatto, specialmente l'ST Basic, anche se con il GFA basic si può sfruttare una buona parte delle possibilità del GEM.
Noi, nel nostro viaggio alla scoperta del GEM, useremo il C, poiché, a nostro modesto parere, è il linguaggio migliore per poter spremere l'Atari fino all'ultimo Bit.
Attenzione, questo non vuole essere un corso di C per principianti; si richiede una minima conoscenza del C. Potete imparare il C leggendo il corso pubblicato su BIT o leggendo uno tra i tanti libri in commercio; noi vi consigliamo "Linguaggio C" di Kernighan & Ritchie, gli inventori del C. Il libro è del Gruppo Editoriale Jackson.
Il compilatore che usiamo è il Megamax C, ma il programma qui pubblicato viene compilato correttamente anche dall'Alcyon C, cioè il compilatore C fornito dall'Atari nel Development Kit. Se utilizzate altri compilatori probabilmente le modifiche da effettuare sul programma sono minime e in seguito vi diremo cosa dovrete cambiare.

INTRODUZIONE AL GEM

Possiamo definire il GEM un "Ambiente Operativo".
La sigla GEM sta a significare Graphic Environment Manager. Il Gem è simile

ad un sistema operativo: potremmo dire che il GEM sta alla grafica e all'interfaccia utente come un sistema operativo classico sta alle operazioni di INPUT-OUTPUT. Esso offre all'utente un modo standard di utilizzare le risorse del computer. Il GEM agisce da interfaccia tra l'utente e il computer in modo tale che l'utente non si debba preoccupare direttamente di cose come il tipo di monitor usato o di stampante.
Il programmatore non deve neanche preoccuparsi direttamente dei menu o delle icone: basta che dica al GEM una volta per tutte come sono fatti i menu, le icone e tutto il resto. Questo tipo di approccio permette di trasportare programmi addirittura su macchine con sistemi operativi diversi purché sia implementato il GEM: basti pensare al PC IBM.
Il GEM è composto da 2 segmenti principali: l'AES, cioè Application Environment Service, e il VDI o Virtual Device Interface.
Il VDI si occupa della grafica di base; il nome significa Dispositivo di Interfaccia Virtuale. Il termine "virtuale" sta a significare che il programmatore non disegna o scrive direttamente sullo schermo o sulla stampante ma su una "macchina grafica virtuale". È il VDI che si preoccupa poi di tradurre i comandi dell'utente in disegni o parole da stampare sullo schermo, sulla stampante, sul plotter ecc. ecc.
L'AES si preoccupa dell'interazione col mouse, con i menù, con le finestre ecc.. Offre cioè un modo standard di interazione tra programma e utente. Sia il VDI che l'AES sono a loro volta suddivisi in componenti, ma per ora non ce ne occuperemo.

DESCRIZIONE GENERALE DEL PROGRAMMA

Il programma qui presentato formatta il disco. "Bella forza" direte voi "lo so fare anche io scegliendo FORMAT dall'apposito menu".
Quello che non sapete è che se il disco è di buona qualità, è possibile aumentare la capacità del disco formattandolo in modo leggermente diverso dal solito. Si ottengono fino a 50 KBytes in più per un disco singola faccia e fino a 100K per un disco a doppia faccia.
Vi è un prezzo da pagare per tanta grazia: non potrete più copiare il disco su un altro semplicemente sovrapponenedo le icone dei due disk-drive; dovete o utilizzare uno dei tanti copiatori in circolazione oppure aprire il disco, selezionare i files voluti e copiarli sull'altro disco.
ATTENZIONE, è possibile che non riusciate ad aumentare la quantità di Bytes utilizzabili di qualche dischetto: ve ne accorgerete perché, quando tenterete di scrivervi sopra, l'Atari si arrabbierà molto. Pazienza, riformattatelo normalmente; il problema non risiede nel programa: semplicemente il disco non sopporta la maxi-formattazione.

```
/**********************************************************************/
/*                                                                    */
/*    format.c                                                        */
/*                                                                    */
/*    questo programma formatta un disco                              */
/*                                                                    */
/*    si puo' aumentare la capacita' di immagazinamento di un disco   */
/*    (per un disco a singola faccia si puo' giungere a oltre 400K )  */
/*    alterando il numero di tracce e di settori per traccia          */
/*                                                                    */
/*    (C)  Dario Bressanini                  6 Gennaio 1987           */
/*    Versione 1.0                                                    */
/*                                                                    */
/**********************************************************************/

#include "obdefs.h"

/*-------------------------------------------*/
/* queste macro aumentano la protabilita' da */
/* un compilatore ad un altro                */
/*-------------------------------------------*/

#define WORD  int                                   /* 16 bit     */
#define LONG  long                                  /* 32 bit     */
#define BYTE  char                                  /* 8 bit      */
#define VOID  /**/                                  /* niente     */
#define NULL  0
#define ERROR       -1
#define NO_MEMORY -2
#define OK          0

/*--------------------------------------*/
/*  interfaccia col sistema operativo   */
/*--------------------------------------*/

extern  LONG    xbios();
extern  BYTE    *gemdos();

#define Flopwr(a,b,c,d,e,f,g)       xbios(9,a,b,c,d,e,f,g)
#define Flopfmt(a,b,c,d,e,f,g,h,i)  xbios(10,a,b,c,d,e,f,g,h,i)
#define Protobt(a,b,c,d)            xbios(18,a,b,c,d)
#define Malloc(a)                   gemdos(0x48,a)
#define Mfree(a)                    gemdos(0x49,a)

/**********************************************************************/
/*   external                                                         */
/**********************************************************************/

extern    WORD  gl_apid;                /* application ID            */

/**********************************************************************/
/*   variabili globali                                                */
/**********************************************************************/

WORD        gem_handle;                 /* handle del desktop         */
WORD        handle;                     /* vdi_handle                 */

WORD        i,dummy;                    /* variabili dummy            */

WORD        contrl[12];                 /* array di controllo         */
WORD        intin[128];
WORD        ptsin[128];
WORD        intout[128];
WORD        ptsout[128];

WORD        work_in[11];                /* Input di v_opnvwk          */
WORD        work_out[57];               /* Output di v_opnvwk         */

/*--------------------------------------*/
/*   struttura dati necessaria per      */
/*   costruire il DIALOG                */
/*                                      */
/*   e' stata generata dal RESOURCE     */
/*   CONTRUCTION SET                    */
/*--------------------------------------*/

BYTE *strings[] = {
"MAXI_FORMATTATORE",
"By Dario Bressanini",
"Format",
"Esci",
"9 Settori",
"10 Settori",
"Singola faccia",
"Doppia faccia",
"80 Tracce",
"81 Tracce",
"82 Tracce"};
```

```
OBJECT tree1[] = {
-1, 1, 13, G_BOX, NONE, OUTLINED, 0x21100L, 0,0, 69,16,
3, 2, 2, G_BOX, NONE, SHADOWED, 0xFE1101L, 3,12, 63,2,
1, -1, -1, G_BOX, NONE, NORMAL, 0xFE1141L, 0,0, 13,2,
4, -1, -1, G_STRING, NONE, NORMAL, 0x0L, 12,1, 17,1,
5, -1, -1, G_STRING, NONE, DISABLED, 0x1L, 35,1, 19,1,
6, -1, -1, G_BUTTON, 0x7, SHADOWED, 0x2L, 58,3, 8,3,
7, -1, -1, G_BUTTON, 0x5, SHADOWED, 0x3L, 58,7, 8,3,
10, 8, 9, G_BOX, NONE, NORMAL, 0xFF1141L, 3,3, 17,7,
9, -1, -1, G_BUTTON, 0x11, SHADOWED, 0x4L, 1,1, 15,2,
7, -1, -1, G_BUTTON, 0x11, SHADOWED, 0x5L, 1,4, 15,2,
13, 11, 12, G_BOX, NONE, NORMAL, 0xFF1121L, 21,3, 17,7,
12, -1, -1, G_BUTTON, 0x11, SHADOWED, 0x6L, 1,1, 15,2,
10, -1, -1, G_BUTTON, 0x11, SHADOWED, 0x7L, 1,4, 15,2,
0, 14, 16, G_BOX, NONE, NORMAL, 0xFF1131L, 39,3, 17,7,
15, -1, -1, G_BUTTON, 0x11, SHADOWED, 0x8L, 1,1, 15,1,
16, -1, -1, G_BUTTON, 0x11, SHADOWED, 0x9L, 1,3, 15,1,
13, -1, -1, G_BUTTON, 0x31, SHADOWED, 0xAL, 1,5, 15,1};

#define TREE1 0     /* TREE */
#define TRACCE80 14             /* OBJECT in TREE #0 */
#define TRACCE81 15             /* OBJECT in TREE #0 */
#define TRACCE82 16             /* OBJECT in TREE #0 */
#define FORMATTA 5              /* OBJECT in TREE #0 */
#define ESCI 6      /* OBJECT in TREE #0 */
#define PROBOX 1    /* OBJECT in TREE #0 */
#define PROBAR 2    /* OBJECT in TREE #0 */
#define MAXI 3      /* OBJECT in TREE #0 */
#define DARIO 4     /* OBJECT in TREE #0 */
#define SETT9 8     /* OBJECT in TREE #0 */
#define SETT10 9    /* OBJECT in TREE #0 */
#define SINGOLA 11              /* OBJECT in TREE #0 */
#define DOPPIA 12               /* OBJECT in TREE #0 */
#define NUM_OBJ 16          /* numero di oggetti */

OBJECT *tree_addr=tree1;            /* puntatore a OBJECT   */


/***********************************************************************/
/*    inizia():       routine di inizializzazione                      */
/*                                                                     */
/*    return    0     nessun errore                                    */
/*              2     non puo' aprire la workstation                   */
/*              3     non puo' inizializzare l'AES                     */
/*                                                                     */
/***********************************************************************/

WORD
inizia()
{
     if( appl_init() == ERROR )           /* inizializza AES           */
          return 3;

     if( open_vwork() == ERROR )          /* apre virtual workstation  */
          return 2;

     graf_mouse(0,0x0L);                  /* cambia in freccia         */
     return OK;                           /* nessun errore             */
}

/***********************************************************************/
/*   termina(livello)                                                  */
/***********************************************************************/

VOID
termina(livello)
WORD livello;
{
     switch ( livello )                   /* ci sono errori ?          */
     {                                    /* nota : non ci sono break  */
                                          /* tutti i case sono in      */
                                          /* cascata                   */


          case 0:
               v_clsvwk(handle);          /* chiude workstation        */

          case 2:                         /* workstation non aperta    */
               appl_exit();               /* esci                      */

          case 3:                         /* AES non inizializzato     */
               ;
     }
}
/***********************************************************************/
/*   open_vwork():  return 0 == OK , return  == ERROR                  */
/***********************************************************************/

WORD
open_vwork()
{
```

USO DEL PROGRAMMA

Il programma è banale da usare. Dopo averlo fatto partire, avrete sullo schermo (bianco/nero oppure colore in media risoluzione) quello che potete vedere in figura 1.

Il pannello che vedete è chiamato, tecnicamente, un DIALOG (per favore, non usate l'italiano DIALOGO: è orrendo). Sul DIALOG vi sono dei BUTTON (se proprio volete usare l'italiano, non usate il termine BOTTONI, ma TASTI) mediante i quali potete scegliere il tipo di formattazione voluta. La formattazione standard è di 9 settori e 80 tracce; scegliendo 10 settori e/o scegliendo 81 o 82 tracce, potete aumentare la capacità del dischetto.

È necessario qui spiegare molto brevemente come è strutturato un dischetto. Per una spiegazione approfondita vi rimando all'apposito articolo sui segreti del floppy-disk pubblicato in questo stesso numero della rivista.

Un disco è suddiviso in tracce: queste sono concentriche e vengono numerate dall'esterno verso l'interno; le tracce normalmente vanno dalla numero 0 alla numero 79, quindi su un dischetto vi sono 80 tracce. Ogni traccia è divisa radialmente in settori, di solito 9; un settore contiene 512 bytes. Questo tipo di formattazione è stata usata dall'ATARI per poter essere compatibile con la formattazione IBM: infatti i nostri drives riescono a leggere i dischi IBM da 3 pollici e mezzo. È possibile però utilizzare 10 settori per traccia e quindi ottenere 0.5K in più per ognuna di esse. È possibile anche creare tracce in più: ovviamente, poiché le tracce sono concentriche, le tracce in più saranno dei cerchi molto piccoli, vicini al centro del disco. Se i cerchi sono troppo piccoli, vi possono essere dei problemi di densità di informazione; l'ATARI se ne accorge e vi avverte. Ma non è tutto: non è possibile avere FISICAMENTE 84 o 85 traccie poiché la testina del disk drive arriva a fine corsa, quindi non tentate di modificare il programma per ottenere numeri di tracce assurdi.

Di solito, ma non sempre, si riesce a formattare il disco con 81, 82 e perfino 83 tracce. Già con 83 tracce la testina è messa a dura prova ed è per questo che abbiamo deliberatamente escluso dal programma questa possibilità.

UTILIZZO DELLE MACRO DI PORTABILITÀ

Cosa sono le macro?

Il C ha la possibilità di sostituire, in fase di compilazione, una qualsiasi sequenza di simboli con un'altra. Ciò si può fare attraverso l'istruzione:

#define nome1 nome2

Tutte le volte che il compilatore trova la sequenza di simboli 'nome1' la sostituisce con la stringa 'nome2'. Tutto ciò può servire a definire delle costanti simboliche. Tutto ciò vi sarà più chiaro se osservate il programma: parole come ER-

ROR o OK comunicano di più e rendono più leggibile il programma rispetto ai corrispondenti −1 e 0. Quando il compilatore trova ERROR, subito lo sostituisce con −1.
In testa al programma sono definite delle "MACRO" utili per permettere la portabilità di un programma da un compilatore ad un altro. Vengono definite le parole BYTE, WORD e LONG. Esse indicano tutte quantità con segno (positivo o negativo) con la convenzione che WORD indica un intero a 16 bit, BYTE a 8 bit e LONG un intero a 32 bit.
Spiego brevemente l'uso: se in un programma ho bisogno della variabile intera a 16 bit chiamata "pippo" posso dichiarare:
int pippo;
infatti, il Megamax C (che noi usiamo) utilizza interi a 16 bit.
Se però voglio compilare lo stesso programma con, ad esempio, il Lattice C, il programma potrebbe (e sottolineo POTREBBE) non funzionare perfettamente poiché il Lattice C usa gli interi a 32 bit.
Se invece dichiaro
WORD pippo;
tutto torna a funzionare: infatti all'inizio del programma vi sarà la macro
define WORD int
e quindi il compilatore Megamax C, anzi il preprocessore, sostituirà WORD con int ogni volta che appare.
Se invece uso il Lattice C, dovrò usare la macro:
define WORD short
poiché "short", per il Lattice C, è una quantità a 16 bit. Tutto chiaro fino ad ora? Bene, proseguiamo.

COMMENTO AL PROGRAMMA

Non commenteremo tutto il programma e la teoria su cui si basa: per far ciò occorrerebbero decine di pagine. Spiegheremo le cose principali e basilari, le prime cose da sapere sul GEM.
La prima istruzione dice al compilatore di INCLUDERE in quel punto, il file OBDEFS.H; torneremo un'altra volta a parlare di questo file. Per ora vi basti sapere che nel file sono contenute definizioni di "oggetti" del GEM: da qui il nome OBject DEFinitionS.
Qualche riga dopo vi è una dichiarazione "extern" che deve essere SEMPRE presente nei vostri programmi. Seguono poi le dichiarazioni di alcune variabili e di 7 array. Gli array denominati "contrl intin ptsin intout ptsout", devono essere SEMPRE presenti nei vostri programmi che sfruttano il GEM altrimenti il vostro linker si arrabbierà non poco. Tutte le comunicazioni con il VDI avvengono tramite questi array ed il VDI si aspetta di trovarli nel vostro programma.
Le variabili chiamate "handle" e "gem_handle" sono 2 "HANDLE": sono cioè due identificatori che indicano al GEM su che dispositivo sta scrivendo o disegnando. Il VDI, cioè la parte del GEM

```
WORD i,pxy[4];

     gem_handle = graf_handle( &i , &i , &i , &i );
     handle=gem_handle;                              /* vdi handle            */

     for( i=0 ; i<10 ; work_in[i++] = 1 )
          ;                                          /* riempie l'array       */
     work_in[10]=2;                                  /* coordinate raster     */
     v_opnvwk(work_in,&handle,work_out);

     if ( handle == 0 )
          return ERROR;                              /* errore in v_opnvwk    */

     return OK;
}

/*************************************************************************/
/*   main : qui inizia l'esecuzione                                      */
/*************************************************************************/

VOID
main()
{
WORD livello;                                  /* livello  inizializzazione  */

     if (!( livello = inizia()))               /* 0 == nessun errore         */
          programma();                         /* esegui programma           */
     termina(livello);                         /* termina secondo "livello'  */
}

/*----------------------------------------*/
/*          define                        */
/*----------------------------------------*/

#define    MAGIC      0x87654321L
#define    VIRGIN     0xE5E5
#define    ILEAVE     1
#define    NOLOAD     0
#define    RANDOM     0x1000000L
#define    BOOTSECT   1
#define    FAT1SECT   2
#define    FAT2SECT   7
#define    TRACK0     0
#define    SIDE0      0

/*------------------------------------------*/
/*    routines per manipolare gli oggetti   */
/*------------------------------------------*/

/*------------------------------*/
/*        do_obj                */
/*------------------------------*/
VOID
do_obj(tree, obj, bit)          /* setta il singolo bit in ob_state     */
OBJECT  *tree;
WORD     obj, bit;
         (tree+obj)->ob_state |= bit;
}

/*------------------------------*/
/*        undo_obj              */
/*------------------------------*/
VOID
undo_obj(tree, obj, bit)        /* cancella il singolo bit in ob_state  */
OBJECT  *tree;
WORD     obj, bit;
{
         (tree+obj)->ob_state &= ~bit;
}

/*------------------------------*/
/*        sel_obj               */
/*------------------------------*/
VOID
sel_obj(tree, obj)                  /* seleziona l'oggetto */
OBJECT  *tree;
WORD     obj;
{
         do_obj(tree, obj, SELECTED);
}

/*------------------------------*/
/*        desel_obj             */
/*------------------------------*/
VOID
desel_obj(tree, obj)            /* deseleziona l'oggetto */
OBJECT  *tree;
WORD     obj;
{
```

```
        undo_obj(tree, obj, SELECTED);
}

/*----------------------------------*/
/*          statep                  */
/*----------------------------------*/

WORD
statep(tree, obj, bit)
OBJECT      *tree;
WORD        obj;
WORD        bit;
{
     return ( ((tree+obj)->ob_state & bit) != 0);
}

/*----------------------------------*/
/*          selectp                 */
/* controlla se un oggetto e' sele-*/
/* zionato                          */
/*----------------------------------*/

WORD
selectp(tree, obj)
OBJECT      *tree;
WORD        obj;
{
     return statep(tree, obj, SELECTED);
}

/*----------------------------------*/
/*          get_xywh                */
/*----------------------------------*/

VOID
get_xywh(tree, obj,x,y,w,h )        /* ritorna x,y,w,h dell'oggetto     */
OBJECT  *tree;
WORD     obj;
WORD     *x,*y,*w,*h;
{
     objc_offset(tree, obj, x, y);
     *w = (tree+obj)->ob_width;
     *h = (tree+obj)->ob_height;
}

/***********************************************************************/
/*   programma()                                                     */
/***********************************************************************/

BYTE *alert="[2][|Formattazione fallita|Controlla il disco|][  OK  ]";
BYTE *alert1="[3][|Errore in scrittura|][  OK  ]";

programma()
{
WORD button;                           /*      bottone uscita alert  */
WORD devno=0;
WORD ret;
WORD x,y,w,h;
WORD exit_obj;
WORD tracce,settori,side;              /* parametri per format       */

    prepara();
    form_center(tree_addr, &x, &y, &w, &h);
    form_dial(0,0,0,0,0,x,y,w,h);
    form_dial(1,(x+w)/2,(h+y)/2,0,0,x,y,w,h);
    for (;;)
    {
        graf_mouse(0,0x0L);                 /* cambia in freccia       */
        tree1[PROBAR].ob_width = 0;
        objc_draw(tree_addr,ROOT,MAX_DEPTH,x,y,w,h);
        exit_obj = form_do(tree_addr,0) & 0x7FFF ;
        if (exit_obj == ESCI )
        {
            form_dial(2,(x+w)/2,(h+y)/2,0,0,x,y,w,h);
            form_dial(3,0,0,0,0,x,y,w,h);
            return ;
        }
        graf_mouse(2,0x0L);              /* cambia in ape              */

        trova_param(&tracce, &settori, &side);
        ret = format(tracce,settori,side,devno);
        if ( ret == NO_MEMORY )
        {
         form_alert(1,"[2][Memoria insufficiente|per formattare][ ESCI ]");
         return;
        }
        if ( ret == ERROR )
            form_alert(1,alert);
        else
```

che si occupa della grafica di base, ha la capacità di disegnare sullo schermo, sul plotter, su una stampante ecc... È necessario quindi indicare al VDI su che cosa deve disegnare (il termine tecnico è WORKSTATION).
Quando apriamo una workstation, in che modo lo vedremo poi, indichiamo al GEM (VDI) su cosa vogliamo disegnare; il GEM ci assegnerà un numero chiamato HANDLE che dovremo utilizzare ogni volta che vorremo disegnare, scrivere e così via.
Questo numero nel nostro programma, è assegnato alla variabile che, con poca fantasia, ho chiamato "handle".
Gem_handle è l'handle associato al desktop.
Subito dopo potete asservare la inizializzazione di vari array; questi definiscono il DIALOG che compare sullo schermo. Il significato dei vari numeri che vedete è troppo complesso per poter essere spiegato ora; sarà l'argomento di una prossima puntata.
Di seguito vi sono le routines inizia(), termina(), open_(work)() e main(). Queste routines costituiscono un corpo unico e mostrano il modo corretto di inizializzare il GEM, se non usate le finestre. Main() chiama inizia(); se tutto è OK chiama la routine programma() e in seguito chiama termina().
Potete prendere di peso queste funzioni e utilizzarle per cominciare a programmare in GEM; non dovete far altro che scrivere una funzione di nome programma() e inserirvi dentro quello che volete. La struttura di queste funzioni è molto lineare.
La funzione inizia() chiama appl_init(); questa funzione si preoccupa di inizializzare correttamente l'AES; se questa ritorna −1 (cioè ERROR dato che ho definito #define ERROR −1) significa che qualche cosa è andato storto.
La funzione open_work() inizializza il VDI e apre una WORKSTATION, cioè indica al GEM su che dispositivo deve scrivere e disegnare, in questo caso lo schermo. Attenzione, appl_int() è una funzione del GEM mentre open_vwork() è nel listato ed è composta di più istruzioni.
Descriviamo brevemente 2 funzioni che probabilmente non incontreremo più: format() e make_boot().

FORMAT(TRACCE, SETTORI, SIDE, DEVNO)

Questa funzione si occupa di formattare tutte le tracce del disco: vengono passati come parametri il numero di tracce, il numero di settori per traccia, il tipo di disco (singola/doppia faccia) e il drive su cui formattare il disco. La prima istruzione richiede memoria al sistema operativo per poter formattare il disco. Se il puntatore al blocco di memoria non è nullo, cioè se vi è abbastanza memoria, la funzione prosegue formattando in sequenza tutte le tracce. Questa operazione viene

eseguita mediante una chiamata alla funzione Flopfmt() dell'XBIOS.
Vengono passati a questa funzione:
buff: è il puntatore al blocco di memoria richiesta prima
filler: un LONG non usato
devno: è il numero del disk-drive, 0 = disco A, 1 disco B ecc...
settori: numero di settori per traccia (di solito 9).
i :numero della traccia (0..79,80,81)
0 / 1: numero della facciata del disco
ILEAVE :interleave factor (vedi articolo sul floppy disk)
MAGIC :numero "magico" che DEVE essere 0×87654321L
VIRGIN :numero con cui si riempiono i settori. Per le prime due tracce è uguale a 0, poiché questo inizializza propriamente le FAT e la DIRECTORY. Per le altre tracce è 0×E5E5.
La funzione ritorna il valore 0 se tutto è OK. Nota per gli accademici della crusca: il verbo "ritornare" per i programmatori ammette complemento oggetto!
Se tutte le tracce sono OK, viene rilasciata la memoria e viene ritornato OK, altrimenti la funzione format() ritorna ERROR.

MAKE_BOOT (TRACCE, SETTORI, SIDE, DEVNO)

Questa funzione costruisce il Boot_sector (settore di boot) del disco e inizializza le FAT (File Allocation Table). Non è compito di questo articolo spiegare cosa è il Boot sector e cosa sono le FAT; vi basti sapere che il boot sector è un settore speciale del disco su cui sono scritte informazioni indispensabili al corretto funzionamento del disco: ad esempio il numero di settori totali, il numero di settori per traccia e il tipo di disco (singola/doppia faccia). Le FAT sono tabelle che permettono al sistema operativo di localizzare fisicamente ogni file sul dischetto.
Il funzionamento è semplice: attraverso Protobt() chiede al sistema operativo di costruire una immagine in memoria di un boot sector standard; successivamente modifica questa immagine inserendovi le informazioni relative al disco che abbiamo precedentemente formattato in modo non standard.
Queste informazioni sono quelle a cui accennavamo prima: numero di settori totali, settori per traccia e tipo di disco.
A questo punto il boot sector viene scritto su disco con la funzione Flopwr().
In seguito vengono inizializzate le FAT e viene rilasciata la memoria. Se le operazioni di scrittura, tramite Flopwr, sono state effettuate senza problemi, la funzione ritorna OK, altrimenti ritorna ERROR.
Bene, per questo numero ci sembra più che sufficiente il malloppo che vi abbiamo finora esposto. Il listato che accompagna l'articolo va battuto con molta cura, perchè spazi, punti e virgola e "pipe" sono molto importanti negli "script" in

```
          {
              ret = make_boot(tracce,settori,side,devno);
              if ( ret == ERROR )
                  form_alert(1,alert1);
          }
          desel_obj(tree_addr,FORMATTA);

}

/*----------------------------------------*/
/*  prepara il DIALOG                     */
/*----------------------------------------*/

VOID
prepara()
{
WORD i;

    for (i=0; i<= NUM_OBJ ; i++ )
        rsrc_obfix(tree_addr,i);

    tree1[TRACCE80].ob_spec = strings[8];        /* inserisce le stringhe */
    tree1[TRACCE81].ob_spec = strings[9];
    tree1[TRACCE82].ob_spec = strings[10];
    tree1[FORMATTA].ob_spec = strings[2];
    tree1[ESCI    ].ob_spec = strings[3];
    tree1[MAXI    ].ob_spec = strings[0];
    tree1[DARIO   ].ob_spec = strings[1];
    tree1[SETT9   ].ob_spec = strings[4];
    tree1[SETT10  ].ob_spec = strings[5];
    tree1[SINGOLA ].ob_spec = strings[6];
    tree1[DOPPIA  ].ob_spec = strings[7];

    sel_obj(tree_addr,SETT9);                    /* oggetti di default    */
    sel_obj(tree_addr,SINGOLA);
    sel_obj(tree_addr,TRACCE80);
}

/*----------------------------------------*/
/*  trova_param                           */
/*  trova il numero di tracce,settori e */
/* il tipo di formattazione scelto        */
/*----------------------------------------*/

VOID
trova_param(tracce,settori,side)
WORD *tracce;
WORD *settori;
WORD *side;
{
    if (selectp(tree_addr,TRACCE80))
        *tracce = 80;
    if (selectp(tree_addr,TRACCE81))
        *tracce = 81;
    if (selectp(tree_addr,TRACCE82))
        *tracce = 82;

    if (selectp(tree_addr,SETT9))
        *settori = 9;
    else
        *settori = 10;

    if (selectp(tree_addr,SINGOLA))
        *side = 0;
    else
        *side = 1;
}
/*--------------------------------------------*/
/* set_prog()                                 */
/* questa routine si occupa della barra che */
/* si riempie mano a mano che si formatta    */
/* il disco                                   */
/*--------------------------------------------*/

VOID
set_prog(value,maxc)
WORD value,maxc;
{
WORD wold,wnew;
WORD x,y,w,h;

    wold = tree1[PROBOX].ob_width -1;       /* lunghezza della barra */
    wnew = wold + 1;
    if (maxc)
    {
        wnew =  ((LONG) value * (LONG) wnew) /maxc ;
        if (wnew < 1)
            wnew = 1;
    }
    wold = tree1[PROBAR].ob_width;
    tree1[PROBAR].ob_width = wnew ;
    if ( value )
```

```
{
     get_xywh(tree_addr,PROBAR,&x,&y,&w,&h);
     x += wold ; w = w-wold+1 ;
     objc_draw(tree_addr,ROOT,MAX_DEPTH,x,y,w,h);
}
/*-------------------------------------*/
/*          format()                   */
/* formatta il disco                   */
/* ritorna ERROR se vi sono errori     */
/* e NO_MEMORY se non vi e' sufficiente */
/* memoria per formattare              */
/*-------------------------------------*/


WORD
format(tracce,settori,side,devno)
WORD tracce;                        /*    numero tracce          */
WORD settori;                       /*    settori per traccia    */
WORD side;                          /*    singola/ doppia faccia */
WORD devno;                         /*    drive A: o B:          */
{
WORD i;                             /*    conta le traccie       */
BYTE *buf;                          /*    buffer per formattare  */
LONG filler;
LONG succ = OL;

     buf = Malloc(10000L);          /*     alloca memoria buffer */
     if (buf == NULL)               /*     poca memoria          */
         return ( NO_MEMORY );

     for (i=0 ; i<tracce ; i++)     /*     formatta le traccie   */
     {
         succ += Flopfmt(buf, filler, devno, settori, i,0,
                         ILEAVE, MAGIC,i<2 ? 0 : VIRGIN);

         if (side != 0)             /*  lato 1                   */
         succ += Flopfmt(buf, filler, devno, settori, i,1,
                         ILEAVE, MAGIC,i<2 ? 0 : VIRGIN);

         set_prog(i+1,tracce);      /* ingrandisci la barra      */
         if (succ)                  /* se vi e' un errore esci   */
             break;

     }
     Mfree(buf);                    /*     rilascia la memoria   */

     if ( succ )
         return ( ERROR );
     else
         return ( OK );
}
/*-------------------------------------*/
/*          make_boot()                */
/* costruisce il boot sector del disco */
/* ritorna ERROR se vi e' un errore    */
/*-------------------------------------*/
WORD
make_boot(tracce,settori,side,devno)
WORD tracce;                        /* numero di tracce          */
WORD settori;                       /* numero di settori         */
WORD side;                          /* singola o doppia faccia   */
WORD devno;                         /* drive A: o B:             */
{
WORD tot_sect=settori*tracce;       /* totale settori            */
BYTE *buf;                          /* buffer per il boot_sector */
WORD i;
LONG filler,succ;

     buf = Malloc(512L);                    /* buffer per boot sect  */
     for(i = 0; i < 512; *(buf + i++) = 0)  /* azzera boot sector    */
         ;
     Protobt(buf, RANDOM, side + 2 , NOLOAD);
     *(buf + 0x13) = (tot_sect + (side ? tot_sect : 0)) & 0xff;
     *(buf + 0x14) = ((tot_sect + (side ? tot_sect : 0)) >> 8) & 0xff;
     *(buf + 0x18) = settori;

     succ = Flopwr(buf,filler,devno, BOOTSECT, TRACK0, SIDE0, 1);

     for(i = 0; i < 512; *(buf + i++) = 0)  /* azzera buffer         */
         ;
     *(buf) = 0xf7;                         /*     costruisce le FAT */
     *(buf + 1) = 0xff;
     *(buf + 2) = 0xff;
     succ += Flopwr(buf,filler,devno, FAT1SECT, TRACK0, SIDE0, 1);
     succ += Flopwr(buf,filler,devno, FAT2SECT, TRACK0, SIDE0, 1);

     Mfree(buf);                            /* rilascia memoria      */
     if (succ)
         return ( ERROR );                  /* errore in flopwr      */
     else
         return ( OK );
```

C. Buon lavoro!

QUANDO IL GEM FINI' IN TRIBUNALE

Fino all'ottobre 1985 era il cavallo di battaglia della Digital Research, la società americana che negli anni Settanta era riuscita ad affermare in tutto il mondo il CP/M, il sistema operativo per i computer dotati di microprocessore Z80, molto diffusi fino al 1980, anno in cui la IBM presentò il PC basato su 8086. Impostosi come standard industriale e supportato da migliaia di realizzazioni software, è ancor oggi il più diffuso al mondo dopo l'MS-DOS. Il GEM aveva le carte in regola per sfondare: facile da usare, potente e versatile, trasformava qualsiasi programma di produttività personale sotto MS-DOS in un sistema "user-friendly", semplice da imparare e molto più veloce da utilizzare. Il segreto del GEM era quello di proporsi come ambiente operativo a icone, simile a quello di Apple Macintosh, con un desktop principale, menu pull-down, uso intensivo del mouse e del linguaggio simbolico.

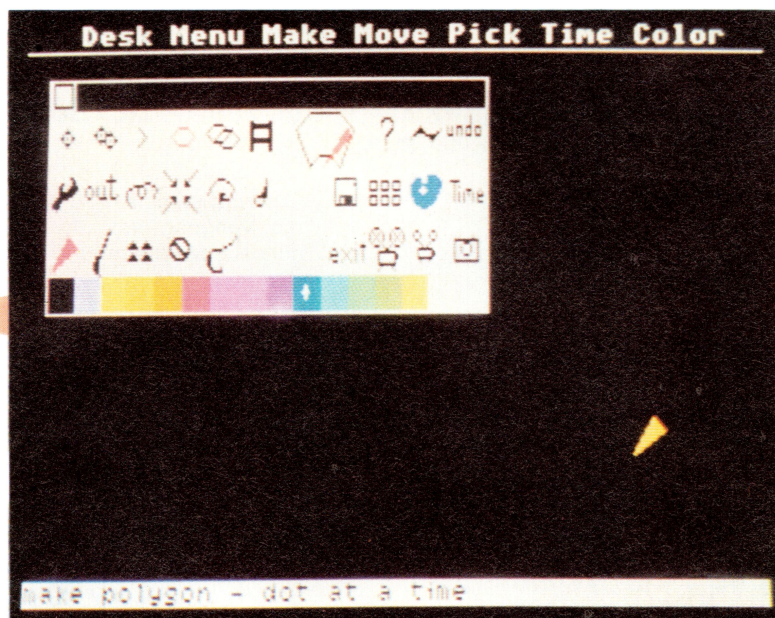
In ambiente MS-DOS il GEM aveva creato subito interesse non solo per le sue caratteristiche intrinseche, ma anche per i programmi che erano stati commercializzati insieme al sistema operativo.

La comparsa dell'ATARI ST, prima con il 520 e poi con il 1040, aveva sicuramente aumentato le credenziali del GEM, che era responsabile del grande successo commerciale delle macchine ST. Ed è forse stato proprio questo sconfinamento in un settore considerato intoccabile dai seguaci della mela a far scattare da parte della Apple una denuncia contro la DRI per violazione del copyright sull'interfaccia a icone. I legali della casa di Cupertino, sostenevano che il GEM era un rifacimento del sistema di Macintosh, e danneggiava le vendite Apple. Timorosa di imbarcarsi in una costosa causa la Digital Research verso la metà di settembre del 1985 si impegnò a ritirare dal mercato tutte le versioni del GEM in commercio, e a modificare sostanzialmente l'aspetto della scrivania. Come risarcimento danni la Apple chiese una notevole somma di denaro e la disponibilità della DRI a collaborare con Apple per lo sviluppo di pacchetti applicativi su Macintosh.

Un anno dopo le vicende giudiziarie del GEM, la Microsoft aveva quasi terminato il Beta-test di Windows, il nuovo ambiente operativo per computer MS-DOS che si è oggi imposto in tutto il mondo come standard per applicazioni grafiche su PC. In effetti Windows non presenta vantaggi rilevanti rispetto al GEM Desktop, e inizialmente ha offerto al pubblico programmi comparabili come prestazioni al Paint e al Write della DRI. Tuttavia, la fortuna di Windows è legata al nome e alla forza commerciale della Microsoft, oggi leader nel software per PC IBM e compatibili.

AEGIS ANIMATOR: ANIMAZIONE FACILE

UN PROGRAMMA PER LA REALIZZAZIONE DI IMMAGINI IN MOVIMENTO INCREDIBILMENTE SEMPLICE DA USARE. L'IDEALE PER CHI SI VUOLE TUFFARE NELLE ANIMAZIONI SENZA PERDERE TEMPO SU VOLUMINOSI MANUALI.

DI PAOLO GALVANI

Preparare una sequenza di immagini in movimento non è mai stata cosa troppo facile ed i migliori programmi del genere si sono spesso rivelati così complicati da scoraggiare presto anche il più incallito degli appassionati. Numerosi comandi, lunghe sequenze da memorizzare e il tempo da dedicarvi consigliavano di rivolgersi ai videogiochi.

Oggi è arrivato Aegis Animator, un programma di animazione che rende addirittura superfluo il manuale di istruzioni. Tutti i comandi sono ben chiari e ordinati nei soliti menu pull-down cosicché la creazione di sequenze diventa elementare. Paradossalmente la cosa più difficile non è preparare la sequenza delle immagini, ma crearle. Non è certo più difficile che disegnare con un comune programma di paint, ma questo vi può dare l'idea di quanto sia facile lavorare con Aegis Animator. Il programma occupa in memoria poco più di 100K lasciano così ampio spazio alle nostre elaborazioni. Alla schermata iniziale il video si presenta completamente nero con i menu pull-down nella barra superiore e il fast menu in alto a sinistra. Questo è un menu che può essere visualizzato o meno, contenente le funzioni più semplici che possono essere selezionate senza "tirare giù" il menu a discesa.

"Project" è il menu di base, quello con cui si inizia la preparazione delle immagini. Qui troviamo "New script", che consente di passare ad un'altra immagine, e "Storage", che accede al drive per caricare sequenze già memorizzate. Importantissima l'opzione "Color", tramite la quale possiamo modificare a piacimento la tavolozza di colori originale variando luminosità e saturazione di ognuno dei trentadue colori disponibili in bassa risoluzione. Sempre in questa opzione troviamo i comandi per selezionare un range di colori e il comando "Cycle" per

ottenere il rapido e continuo cambiamento dei colori della nostra immagine. Sempre nel menu "Project" abbiamo "Undo", "Fast menu" (per visualizzarlo o farlo sparire), "Storyboard" (per vedere contemporaneamente e spostarsi nelle nove pagine grafiche di Aegis Animator), "Time" (per regolare la velocità del movimento delle immagini) e "Status" (che indica la memoria a disposizione).

sti due modi di visualizzazione forniscono un'ampia versatilità al programma. Per stabilire le dimensioni degli oggetti ci si sposta, partendo dal centro della figura, verso l'esterno finché l'oggetto non avrà raggiunto le dimensioni desiderate. Con l'opzione "Clone" è possibile duplicare uno qualsiasi degli oggetti presenti sullo schermo e con "Destroy" cancellarlo.

CREARE UN OGGETTO

La creazione di un oggetto avviene in modo molto semplice. Il menu "Create" contiene una serie di oggetti già pronti da utilizzare, ovvero linee, poligoni (da costruirsi manualmente o già pronti a quattro lati), cerchi e stelle. I cerchi, le linee e i poligoni da costruire possono essere visualizzati "pieni", cioè colorati, oppure semplicemente con le linee esterne. Que-

IL MOVIMENTO

Creare il movimento di un oggetto è molto semplice grazie ai menu "Move" e "Select". Possiamo scegliere tra due diversi modi di procedere: creare tutti gli oggetti e poi muoverli singolarmente o a gruppi, oppure animare le figure mano a mano che le costruiamo.
Ognuno di voi adotterà naturalmente il metodo che gli sembrerà più comodo.

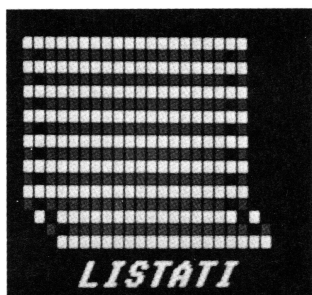
Prima di preparare l'animazione bisogna selezionare la figura da muovere. Tramite il menu "Select" possiamo scegliere un punto, tutti i punti presenti sullo schermo, un segmento, tutti i segmenti, un poligono, tutti i poligoni oppure tutti gli oggetti della pagina grafica.
Una volta effettuata la selezione si decide che tipo di movimento dovrà fare l'oggetto. Possiamo spostarlo orizzontalmente (sideways), portarlo dal centro dello schermo verso il bordo (out) e viceversa (in), ruotarlo in piano (plane), sull'asse x e sull'asse y, modificarne le dimensioni, creare un percorso sul quale muoverlo (path), cambiarne il colore (change color), cambiarlo da colorato a vuoto e viceversa (change type) e cambiarne la forma (morph).
Tutto questo in maniera molto semplice, solamente selezionando l'opzione desiderata e portandoci sull'oggetto che muoveremo con il mouse "agganciandolo" con un uncino che appare al posto del cursore.

RIVEDERE UNA SEQUENZA

Per rivedere una sequenza creata utilizzeremo il menu "Time". Con "Replay tween" possiamo far partire la sequenza del movimento di una sola figura, con "Replay all" rivediamo il filmato completo. Comoda e spesso utile è "Play loop" che fa ripartire la sequenza ogni volta che giunge alla fine.
In "Ghost mode" tutti gli oggetti sullo schermo diventano trasparenti, in modo da studiare meglio i movimenti dei singoli oggetti nel contesto globale.

CONCLUSIONI

È veramente incredibile la semplicità d'uso di questo programma che permette di ottenere risultati finali che non mancheranno di stupire i vostri amici. La cosa che più si può imputare a Aegis Animator è la mancanza di precisione. I movimenti vengono creati a occhio, e se questo riduce di molto le difficoltà è vero anche che in molti casi la precisione è essenziale. Ma del resto questo programma si rivolge a chi vuole "smanettare" un po' con Atari, non certo a chi ci deve lavorare.
Insomma, chi si vuole divertire con animazioni create in proprio, e con risultati niente male, senza ricorrere ad approfonditi studi, deve rivolgersi ad Aegis Animator, il programma per l'animazione facile.

Programma: Aegis Animator
Produttore: Aegis Development
Distributore: per ora non distribuito in Italia, viene venduto in Svizzera
Computer: Atari serie ST, monitor a colori
Prezzo: circa 160 Franchi Svizzeri (170.000 lire)

BUSIGRAF

SPAZIO BASIC

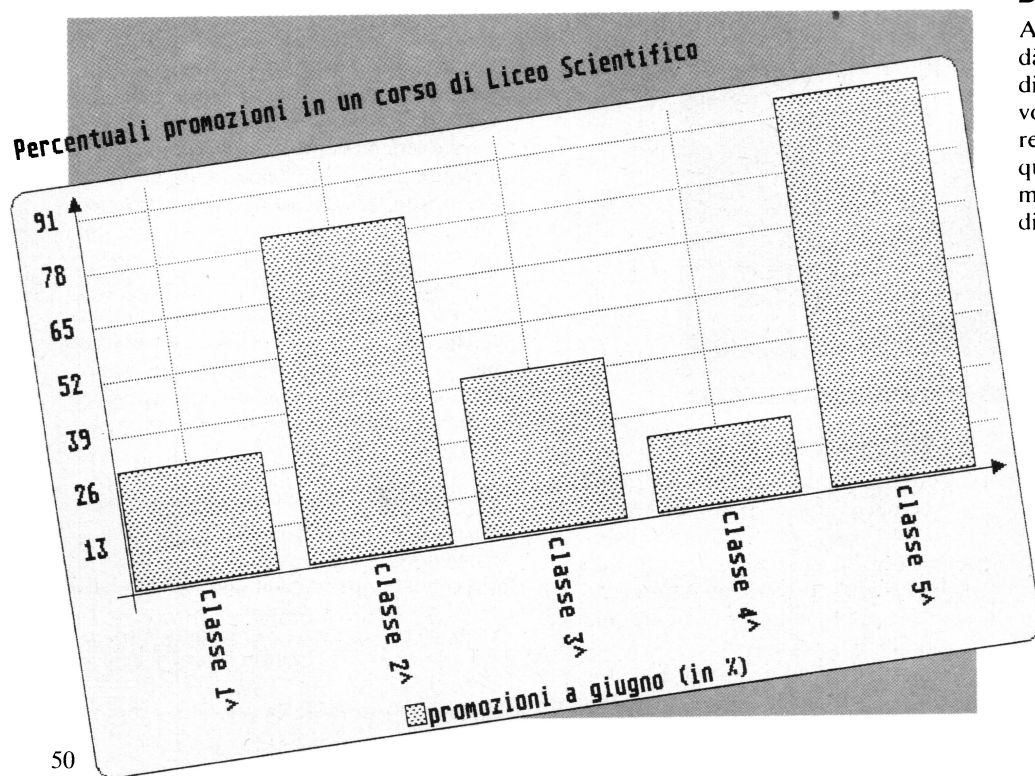
UN PROGRAMMA IN ATARI BASIC PER PRODURRE GRAFICI DI TUTTI I GENERI CON IL 520 IN ALTA RISOLUZIONE

DI MARCELLO MORCHIO

Dopo aver portato a casa il "bestione" era assolutamente necessario tirarci fuori qualcosa di utile, se non altro per non fare temere a chi ha pagato (genitori nel mio e nel rimanente 70% dei casi) di aver speso inutilmente i propri soldi.
Domanda: cosa si adatta meglio a mostrare la possanza di un 520?
Risposta: un programma di grafica, magari utile, oltre che bello.

Ed ecco BUSIGRAF.
Il programma permette la memorizzazione di sei (o più) classi di dati, le quali possono essere rappresentate mediante diagrammi a torta, istogrammi (orizzontali, verticali, multipli, sovrapposti), diagrammi cartesiani, anche multipli, e diagrammi ad aree, naturalmente potendo sovrapporre più diagrammi (istogrammi+aree e istogrammi+cartesiani). I dati possono poi essere manipolati in vari modi: si possono ordinare alfabeticamente e numericamente, in ordine crescente e decrescente. Si possono infine memorizzare e caricare da disco files contenenti singole classi e cancellarli quando non ci sono più utili. Una grande varietà di operazioni quindi, ma di facile impiego, essendo tutto basato su una struttura di menu che guida l'utente durante tutte le operazioni. Lo scaling è completamente automatizzato come anche l'introduzione di legende esplicative dei grafici.

BUSIGRAF IN PRATICA

Al RUN appare l'INPUT MENU, che dà accesso evidentemente alle operazioni di inserimento dati. Per cominciare a lavorare è naturalmente necessario inserire dei dati da rappresentare: chiamiamo quindi l'opzione [1] "Nuova classe". Immesso il nome che si vuole dare alla serie di dati (che sarà quello che apparirà nel-

Questa è la rappresentazione grafica più semplice di BUSIGRAF: un istogramma. Al posto degli accessori della Output Window ora c'è l'intestazione del grafico.

```
1000  rem         ________________________________________________
1010  rem         |           I N I Z I A L I Z Z A Z I O N E          |
1020  rem         ------------------------------------------------
1030  clear:classi=5:pi=3.141592:cpu%=classi+1:fullw 2
1040  option base 0:on error goto 6060
1050  dim va(23,classi+1),elem%(classi+1),com$(23,classi+1),nome$(classi)
1060  dim ev$(23,classi)
1070  for n=0 to classi:for j=0 to 23
1080  va(j,n)=0:com$(j,n)="*":ev$(j,n)=" ":next
1090  elem%(n)=0:nome$(n)="*":next
1100  fr$="\xxxxxxxx\ ######.##":fr1$="##) "
1110  cu%=0:ncl%=1:openw 2,0,18,690,440
1120  x%=0:y%=0:i%=640:j%=440:color 1,0,1,1,1
1130  SEZIN:
1140  rem         ____________________________________
1150  rem         |           INPUT     MENU             |
1160  rem         ------------------------------------
1170  gosub ALLW2:gotoxy 0,0:?" ***** BUSUGRAF ****":?
1180  ?"INPUT MENU:":?:?:?"[1]      NUOVA CLASSE DATI"
1190  ?"[2]      AGGIUNGI DATI"
1200  ?"[3]      CORREGGI DATI";tab(35);"[4]      CANCELLA CLASSE"
1210  ?"[5]      OUTPUT NUMERICO"
1220  ?"[6]      OUTPUT MENU":?"[7]      UTILITY":?
1230  ?"classi usate:":if cu%=0 then ?"-Nessuna-":?:goto 1260
1240  for n=0 to cu%-1:?n+1;":";nome$(n);" ";:next:?:?
1250  fl%=0
1260  ?"cosa scegli ?";:gosub ASKNUM
1270  on a% gosub NUOVA,AGGIUNGI,CORREGGI,CANCMEM,NUMSTAMP
1280  if a%=6 then goto SEZOUT
1290  if a%=7 then goto TOOLS
1300  gosub ALLW2:goto SEZIN
1310  NUOVA:
1320  rem         ___________________
1330  rem         | nuova classe dati |
1340  rem         -------------------
1350  if cu%>classi then ?:?"non ci sono piu' classi disponibili":return
1360  input "nome nuova classe";n$
1370  nome$(cu%)=n$:cl%=cu%:cu%=cu%+1
1380  da%=0:gosub RIEMPIMENTO
1390  return
1400  AGGIUNGI:
1410  rem         _________________________
1420  rem         | aggiunta dati a classe |
1430  rem         -------------------------
1440  if cu%=0 then ?:?"non sono state ancora definite classi":return
1450  gosub ASKCLASS:if cl%=-1 then return
1460  da%=elem%(cl%):if da%>23 then ?:?"classe gia' piena":return
1470  gosub RIEMPIMENTO
1480  return
1490  CORREGGI:
1500  rem         __________________
1510  rem         | correzione dati |
1520  rem         ------------------
1530  if cu%=0 then ?:?"non sono state ancora definite classi":return
1540  gosub ASKCLASS:if cl%=-1 then return
1550  fl%=1:gosub NUMSTAMP1
1560  gotoxy 0,16:input "quale modifico (x=ret)";r$:a%=val(r$)
1570  gotoxy 0,16:? spc(70)
1580  if r$="x" or r$="X" then return
1590  gotoxy 0,16:?a%;")";:input "denominazione,valore";com$(a%,cl%),va(a%,cl%)
1600  line input "* =evidenzia [RETURN] =normale ";r$
1610  if r$<>"*" then r$=" "
1620  ev$(a%,cl%)=r$:gosub STAMPAA
1630  gotoxy 0,16:? spc(70):? spc(70):? spc(70):goto 1560
1640  CANCMEM:
1650  rem         __________________
1660  rem         | cancella da mem |
1670  rem         ------------------
1680  ?"CANCELLAZIONE IN MEMORIA:":gosub ASKCLASS:if cl%=-1 then return
1690  for n=cl%+1 to classi:elem%(n-1)=elem%(n)
1700  for j=0 to 23:va(j,n-1)=va(j,n):com$(j,n-1)=com$(j,n):ev$(j,n-1)=ev$(j,n)
1710  next:nome$(n-1)=nome$(n):next:cu%=cu%-1:elem%(classi)=0
1720  return
1730  NUMSTAMP:
1740  rem         __________________
1750  rem         | output numerico |
1760  rem         ------------------
1770  if cu%=0 then ?:?"non sono stati memorizzati dati":return
1780  gosub ASKCLASS:if cl%=-1 then return
1790  NUMSTAMP1:
1800  gosub ALLW2:gotoxy 0,0:?"nome classe:";nome$(cl%)
1810  for a%=0 to elem%(cl%)-1:gosub STAMPAA:next
1820  if fl%=0 then a%=inp(2)
1830  return
1840  RIEMPIMENTO:
1850  rem         ______________________________________
1860  rem         | riempi classe cl% a partire da da% |
1870  rem         --------------------------------------
1880  gosub ALLW2:gotoxy 0,0:fl%=1
1890  gosub NUMSTAMP1
1900  gotoxy 0,16:?"n.";da%;":";
1910  input "denominazione,valore (0,0=exit)";com$(da%,cl%),va(da%,cl%)
1920  if com$(da%,cl%)="0" then goto 1980
1930  line input "* =evidenzia [RETURN] =normale ";r$
1940  if r$<>"*" then r$=" "
1950  ev$(da%,cl%)=r$:a%=da%:gosub STAMPAA
1960  gotoxy 0,16:? spc(70):? spc (70):? spc (70)
1970  da%=da%+1:if da%<24 then goto 1900
1980  elem%(cl%)=da%:return
```

l'output), si passa all'input vero e proprio. Per ogni voce occorre inserire prima la definizione e, separato da una virgola, il valore numerico corrispondente. Alla fine dei dati inserire "0,0" per comunicare al programma di tornare al menu. Se a questo punto ci accorgiamo di avere commesso errori nell'immissione, si può passare al modo CORREZIONE, opzione [3] del menù, o, se si vogliono aggiungere altre voci, al modo AGGIUNGI (opzione [2]). Se poi vogliamo produrre un tabulato con i dati numerici, ecco l'opzione [5]. Passiamo ora alla parte centrale del programma: l'OUTPUT MENU, attraverso l'opzione [6] del menu input.

Le possibilità che ci si offrono a questo punto sono piuttosto varie. Analizziamole con ordine: rispondiamo [1] all'input e, dopo avere inserito il nome o il numero della classe su cui si intende lavorare, ecco apparire un diagramma a torta rappresentante i nostri dati, espressi qui in forma di percentuale. Volendo evidenziarne alcuni, separando uno o più spicchi del diagramma, si deve inserire durante l'input un asterisco alla domanda "evidenzio il dato?".

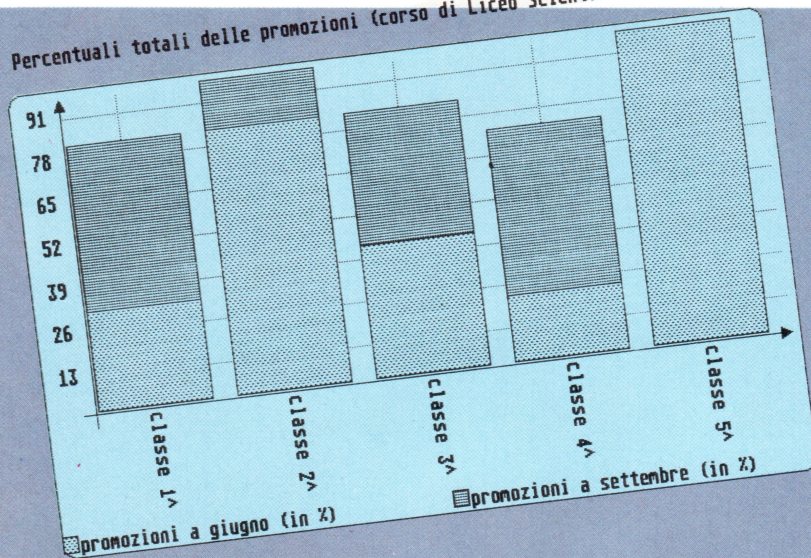
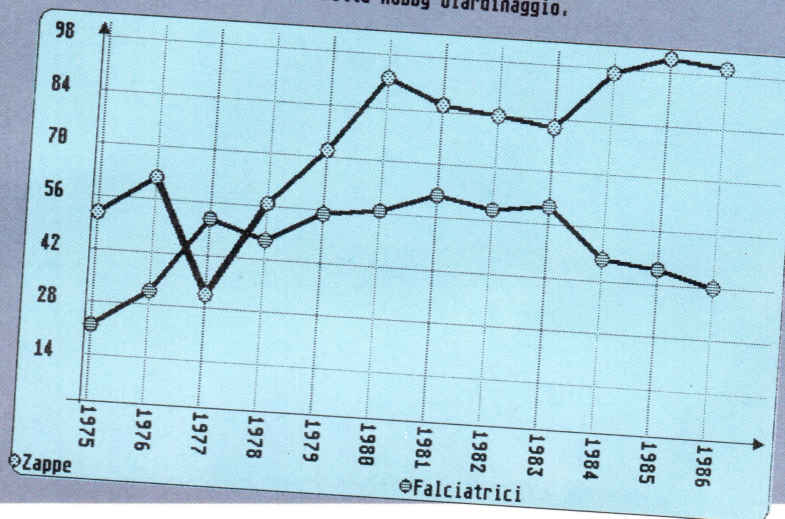
Rispondendo [2] all'output menu, si accede alla tracciatura di istogrammi verticali. Il programma chiede quante classi di dati si vogliono rappresentare e con che tipo di istogrammi (affiancati o sovrapposti). Se vogliamo rappresentare una sola classe possiamo rispondere [RETURN] all'ultima domanda. È quindi necessario comunicare al programma quali classi rappresentare (come ho detto il programma accetta sia il nome che il numero di identificazione), e in che ordine, concludendo con la distanza (in pixels) alla quale andranno posti i gruppi di barre. Chiedere questo dato può sembrare un'inutile complicazione rispetto al chiedere direttamente la larghezza delle barre, ma in realtà ciò risulta più comodo, per l'utente e per il programma. In questo modo se vogliamo che le barre siano tutte unite basterà rispondere "O" all'input, invece di calcolare per tentativi la larghezza necessaria. Inserito il codice della retinatura desiderata per il disegno, ecco apparire il nostro istogramma, con tanto di legenda esplicativa in fondo alla finestra. A proposito: se non lo aveste ancora notato, il programma cancella la Menu Bar in cima allo schermo e i bordi della Output Window, permettendo di riprodurre i grafici su stampante (con ALTERNATE+HELP, per chi ancora non lo sapesse) senza portarsi dietro il marchio del Basic.

Premendo un tasto si torna all'Output Menu. L'alternativa che segue permette la stampa di un istogramma orizzontale. Il procedimento è simile al precedente, a parte il fatto che in orizzontale è possibile rappresentare una sola classe per volta.

L'opzione [4] permette la rappresentazione di diagrammi cartesiani.
Possono essere rappresentate più classi contemporaneamente e il procedimento è uguale a quello usato nella tracciatura di istogrammi multipli. L'unica differenza è che al posto della distanza tra le barre è chiesta la larghezza del tratto. Il grafico rappresenta le linee egualmente larghe e distingue una classe dall'altra attraverso la retinatura dei cerchi disegnati a intervalli regolari sui segmenti.
L'ultima opzione di disegno "singolo", ovvero con un solo tipo di grafico, è la [5]. Essa permette la tracciatura di diagrammi ad aree. Il processo di input è sempre il solito, ma qui occorre segnalare che, proprio perchè i dati sono rappresentati nello stesso ordine col quale sono inseriti, è consigliabile in questo caso inserire per prime le classi che abbiano una media di valori alta, per evitare intersezioni e sovrapposizioni errate delle linee che formano la campitura delle aree. Infatti in questo caso la superficie non è riempita con le normali retinature del GEM, ma con linee a diverse distanze.

Si passa infine alle opzioni di disegno "misto": la [6] rappresenta uno o più istogrammi verticali con sullo sfondo un diagramma ad aree; la [7] rappresenta uno o più istogrammi ai quali è sovrapposta la linea di un diagramma cartesiano. In entrambi i casi, durante l'inserimento dei nomi delle classi da rappresentare, si viene avvertiti che la classe inserita per prima sarà visualizzata con un diagramma ad aree o cartesiano.

Ricordo una cosa che può sembrare ovvia: in tutti i casi in cui si effettuano tracciamenti multipli (sia di un solo tipo che misti) è necessario che il numero dei valori delle varie classi da rappresentare sia lo stesso e che la descrizione dei valori coincida. In caso la descrizione differisca, è visualizzata solo quella della prima classe inserita.
Le opzioni restanti sono funzioni di servizio, che permettono di spostarsi agli altri menu: con la [8] si torna all'Input Menu, con la [9] si accede all'Utility Menu.
Nell'Utility Menu le opzioni sono piuttosto semplici: si possono ordinare i dati di ogni classe in senso crescente o decrescente; si possono ordinare le descrizioni di ogni dato alfabeticamente (anche qui è presente la funzione inversa), e si può infine accedere all'ultimo menu: il menu I/O DISK, che permette appunto la memorizzazione, il caricamento in memoria e la cancellazione di files contenenti ognuno una classe. Il programma riconosce solo files generati da lui stesso o nello stesso formato, e aventi l'estensione 'DAT', che viene aggiunta automaticamente in fase di registrazione e di caricamento e non deve essere quindi inserita da tastiera.

```
1990  STAMPAA:
2000  rem       ________________________________
2010  rem       | stampa item a% in posizione |
2020  rem       --------------------------------
2030  if a%<12 then x%=0:y%=2+a% else x%=20:y%=a%-10
2040  gotoxy x%,y%:n=a%
2050  ?using fr1$;n;:? using fr$;com$(n,cl%);va(n,cl%);ev$(n,cl%):return
2060  ASKCLASS:
2070  rem       _________________________________
2080  rem       | out:  classe da usare in cl% |
2090  rem       ---------------------------------
2100  input "su quale classe vuoi operare";n$:cl%=-1
2110  if val(n$)>0 and val(n$)<=cu% then cl%=val(n$)-1:return
2120  for j=0 to cu%-1:if n$=nome$(j) then cl%=j:j=cu%
2130  next:if cl%=-1 then ?"classe inesistente":return
2140  return
2150  ASKNUM:
2160  rem       __________________
2170  rem       | input 1 numero |
2180  rem       ------------------
2190  a%=inp(2):if a%<48 or a%>57 then goto 2170
2200  a%=a%-48:?"<";a%;">":r$=str$(a):return
2210  SEZOUT:
2220  rem       ------------------------------------------
2230  rem       |           O U T P U T   M E N U        |
2240  rem       ------------------------------------------
2250  gosub ALLW2:gotoxy 0,0:?" **** BUSIGRAF ****":?
2260  ?"OUTPUT MENU:":?:?
2270  ?"[1]      DIAGRAMMA A TORTA"
2280  ?"[2]      ISTOGRAMMI VERTICALI"
2290  ?"[3]      ISTOGRAMMA ORIZZONTALE"
2300  ?"[4]      DIAGRAMMI CARTESIANI"
2310  ?"[5]      DIAGRAMMI AD AREE"
2320  ?"[6]      AREE+ISTOGRAMMI";tab(30);"[7]      CART.+ISTOGRAMMI"
2330  ?"[8]      INPUT MENU":?"[9]      UTILITY":?
2340  ?"classi usate:":if cu%=0 then ?"-Nessuna-":?:goto 2360
2350  for n=0 to cu%-1:?n+1;":";nome$(n);" ";:next:?
2360  ?"cosa scegli ?";:gosub ASKNUM
2370  if a%=3 then r$="or"
2380  if a%=9 then goto TOOLS
2390  if a%=8 then goto SEZIN
2400  xa%=2:ya%=40:yb%=380:xb%=610
2410  on a% gosub AREOGRAMMI,ISTOMULT,BARRE,CARTEMULT,AREE,MISTO1,MISTO2
2420  goto SEZOUT
2430  rem >>>>>>>>> xa%,ya% = coordinate finestra output <<<<<<<<<<
2440  rem >>>>>>>>> xb%,yb% = coords f.o. basso destra   <<<<<<<<<<
2450  AREOGRAMMI:
2460  rem       ________________
2470  rem       | aerogrammi |
2480  rem       --------------
2490  gosub ASKCLASS:if cl%=-1 then return
2500  poke systab+24,1
2510  x%=xa%:y%=ya%:i%=xb%:j%=yb%:color 1,0,1,1,1:gosub VBAR:gosub VRBOX
2520  xc%=xa%+(xb%-xa%)/4:yc%=ya%+(yb%-ya%)/2-33:rg%=(xb%-xa%)/4-4
2530  tot=0:for n=0 to elem%(cl%)-1:tot=tot+va(n,cl%):next
2540  an=0
2550  for n=0 to elem%(cl%)-1:x%=xc%:y%=yc%
2560  an1=va(n,cl%)*3600/tot
2570  if ev$(n,cl%)="*" then ag=(an*pi/1800)+(an1*pi/1800)/2:x%=x%+10*cos
      (ag):y%=y%-10*sin(ag)
2580  color 1,1,1,n+1,2:pcircle x%,y%,rg%,an,an+an1
2590  an=an+an1:next
2600  testo$=nome$(cl%):x%=xa%+(xb%-xa%)/2-8*len(testo$)/2:y%=ya%+16
2610  gosub TEXT
2620  yc%=ya%+48:xc%=xa%+(xb%-xa%)/2:for n=0 to elem%(cl%)-1
2630  x%=xc%:y%=yc%:i%=x%+12:j%=y%-12:color 1,1,1,n+1,2:gosub VBAR
2640  max=va(0,cl%):max$=com$(0,cl%)
2650  pc%=100*va(n,cl%)/tot:testo$=com$(n,cl%)+" "+str$(pc%)+" %"
2660  x%=x%+13:gosub TEXT
2670  yc%=yc%+(yb%-ya%)/13:if n=11 then xc%=xc%+(xb%-xa%)/4:yc%=ya%+48
2680  next:poke systab+24,0
2690  a%=inp(2):return
2700  BARRE:
2710  rem       ________________
2720  rem       | istogrammi |
2730  rem       --------------
2740  gosub ASKCLASS:if cl%=-1 then return
2750  input"che distanza pongo tra i blocchi(in pixels)";dist%
2760  input"retinatura(1-24)";col%
2770  poke systab+24,1
2780  gosub AXESDAT
2790  largh%=(bxw%-(elem%(cl%)*dist%))/elem%(cl%)
2800  if r$="or" then largh%=(byh%-(elem%(cl%)*dist%))/elem%(cl%)
2810  di%=dist%+largh%:incr%=largh%+2:gosub AXES
2820  SUBBARRE:
2830  rem >>> TRACCIATURA BOX <<<
2840  color 1,1,1,col%,2
2850  for n=0 to elem%(cl%)-1
2860  y%=yc%-va(n,cl%)*byh%/max:x%=xc%+n*di%+5:i%=x%+largh%:j%=yc%-5
2870  if r$="or" then x%=xc%+5:y%=yc%-(n+1)*di%-5:i%=x%+va(n,cl%)*bxw%/max:
      j%=y%+largh%
2880  gosub VBAR
2890  next
2900  testo$=nome$(cl%):x%=xa%+(xb%-xa%)/2-8*len(testo$)/2:y%=yb%-16
2910  if ncl%<>1 then x%=xa%+((xb%-xa%)/ncl%+10)*j
2920  if fl%=2 then x%=xa%+((xb%-xa%)/(ncl%+1)+10)*j
2930  i%=x%+12:j%=y%+12:gosub VBAR:y%=j%:x%=i%+2:gosub TEXT
2940  poke systab+24,0
2950  if ncl%<>1 then return
2951  if ncl%<>1 then return
2960  a%=inp(2)·return
```

Saliamo di complessità. Quando abbiamo due classi omogenee di dati che, sommate, acquistano un ulteriore significato, la rappresentazione grafica più adatta è costituita dagli istogrammi sovrapposti.

Se abbiamo dei dati che rappresentano lo svolgersi di un fenomeno nel tempo, il grafico più adatto è il diagramma cartesiano. Se poi i dati sono multipli, tanto meglio: c'è il diagramma cartesiano multiplo.

```
2970  AXESDAT:
2980  rem      ________________________
2990  rem      | disegna assi quotati |
3000  rem      ------------------------
3010  rem >>> CALCOLO COORDINATE ASSI E DIMENSIONI <<<
3020  max=va(0,cl%):max$=com$(0,cl%)
3030  for n=1 to elem%(cl%)-1:if va(n,cl%)>max then max=va(n,cl%)
3040  if len(com$(n,cl%))>len(max$) then max$=com$(n,cl%)
3050  next:s1%=len(str$(max))*12+1:s2%=len(max$)*9+18
3060  xc%=xa%+s1%:yc%=yb%-s2%:bxw%=xb%-xc%-10:byh%=yc%-ya%-10
3070  if r$="or" then xc%=xa%+s2%:yc%=yb%-s1%
3080  bxw%=xb%-xc%-10:byh%=yc%-ya%-10
3090  return
3100  AXES:
3110  rem >>> TRACCIATUTRA ASSI <<<
3120  poke systab+24,0:input"Intestazione ";inte$:poke systab+24,1:gosub ALLW2
3130  x%=xa%:y%=ya%:i%=xb%:j%=yb%:color 1,0,1,1,1:gosub VBAR:gosub VRBOX
3140  i%=0:j%-1:gosub ESTREMI
3150  linef xc%-10+xa%,yc%-ya%,xc%+bxw%+xa%,yc%-ya%
3160  linef xc%+xa%,yc%+10-ya%,xc%+xa%,yc%-byh%-ya%
3170  j%=0:gosub ESTREMI:i%=&haaaa:gosub LINDEF
3180  tp%=max/7:if tp%=0 then tp%=1
3190  i%=0-2700*(r$="or"):gosub BASELINE
3200  for a%=tp% to max step tp%:y%=yc%-(a%*byh%/max)+5:x%=xc%-s1%+5
3210  if r$="or" then y%=yc%:x%=xc%+(a%*bxw%/max)+5
3220  testo$=str$(a%):gosub TEXT
3230  if r$<>"or" then linef xc%+xa%,y%-ya%,xc%+xa%+bxw%,y%-ya%
3240  if r$="or" then linef x%-xa,yc%-ya%,x%,yc%-ya%-byh%
3250  next
3260  i%=2700+2700*(r$="or"):gosub BASELINE
3270  for n=0 to elem%(cl%)-1:x%=xc%+n*di%+incr%:y%=yc%
3280  if r$="or" then x%=xa%+5:y%=yc%-n*di%-incr%
3290  testo$=com$(n,cl%):gosub TEXT
3300  if r$<>"or" then linef x%+xa%,yc%-ya%,x%+xa%,yc%-ya%-byh%
3310  if r$="or" then linef xc%+xa%,y%-ya%,xc%+xa%+bxw%,y%-ya%
3320  next
3325  i%=0:gosub BASELINE:i%=1:gosub SETTYPE:
3330  x%=10:y%=20:testo$=inte$:gosub TEXT
3335  return
```

IL PROGRAMMA

Passiamo ora sinteticamente ad esaminare il programma BASIC di Busigraf, che sarà analizzato approfonditamente nell'apposito spazio (REMarks).

Scorrendo rapidamente il listato si nota che il programma è composto da una serie di piccole routines che si chiamano a vicenda: da quelle di alto livello, che gestiscono i vari menu, a quelle di livello più infimo, che svolgono compiti elementari come chiedere 'Su quale classe vuoi operare?' e comunicare il dato al programma principale. Che non esiste. Mi rendo conto che sto seminando il panico con questa mia ultima affermazione; vediamo di spiegare: un effettivo MAIN PROGRAM, che dovrebbe gestire un ipotetico MAIN MENU, non esiste perchè l'esecuzione parte subito coll'INPUT MENU, che è necessariamente la prima tappa per lavorare con Busigraf. Gli altri menu sono chiamati da apposite opzioni dell'input menu, che viene chiamato a sua volta dagli altri quando necessita. I tre menu principali quindi (Input, Ou-

tput, Utility) sono esattamente sullo stesso piano, proprio come i quattro Desk, File, View, Options del GEM.
Questa configurazione a scatole cinesi, insieme all'abbondanza di REM, rende il programma molto più comprensibile e flessibile, anche grazie alla possibilità offerta dal BASIC Atari di etichettare con nomi mnemonici i numeri di linea, evitando errori derivati da spostamenti di routines o riorganizzazioni del programma in fase di creazione, e permettendo a chiunque di apportare modifiche al listato secondo le proprie esigenze. Basterà infatti aggiungere una voce al menù utility e una breve routine in coda al programma per permettere ad esempio, di calcolare una nuova classe di dati operando sui valori delle precedenti, creando una specie di rozzo spread data sheet con rappresentazione grafica dei dati. Non avrà bisogno di questa opzione chi deve rappresentare dati già esistenti (studenti alle prese con ricerche a cui dare una veste un po' curata), mentre chi dalle spese e dai guadagni vuole ottenere il grafico del ricavo potrà utilmente avvalersene.

N.B.
Un'ultima nota: se, dopo avere definito una classe di dati, vedete apparire nei grafici, tra i nomi o tra i valori numerici, degli strani caratteri o numeri pazzeschi, non prendetevela con Busigraf, ma con i progettisti del Basic Atari, che ogni tanto cambia dei valori nelle matrici di una certa dimensione, almeno nella versione in mio possesso (la prima).

REMarks

1000-1120: Inizializzazione delle variabili 'globali' (descritte nell'apposito riquadro); si stabilisce anche di saltare alla routine 6060 in caso di errore e si apre la Output Window del BASIC oltre i limiti dello schermo, per eliminare tutti i suoi accessori.
1130-1300 (SEZIN): input menu. Vengono rappresentate le opzioni ed eseguite le scelte dell'utente. Notare la chiamata "gosub ALLW2", che cancella dallo schermo tutti gli accessori della Output Window e la chiamata alla routine ASKNUM, che accetta una cifra in INPUT.
1310-1390 (NUOVA): prima opzione dell'input menu. Viene definita una nuova classe di dati. La routine NUOVA si occupa comunque solo di chiedere il nome della nuova classe. Il riempimento è affidato alla routine RIEMPIMENTO.
1400-1480 (AGGIUNGI): aggiunta di dati alla classe indicata. Anche questa routine chiama RIEMPIMENTO che, come vedremo, accetta vari input a seconda della funzione che deve svolgere.
1490-1630 (CORREZIONE): chiede nell'ordine il nome della classe e, visua-

```
3340  SUBCARTE:
3350  rem         ________________________
3360  rem         | diagrammi cartesiani |
3370  rem         ------------------------
3380  i%=largh%:gosub SETWIDTH:color 1,1,1,col%,2
3390  i%=2:j%=2:gosub ESTREMI
3400  x%=xc%+xa%+incr%:y%=yc%-va(c,cl%)*byh%/max-ya%
3410  if ev$(0,cl%)="*" then pcircle x%,y%,largh%+10
3420  for n=1 to elem%(cl%)-1
3430  y1%=yc%-va(n,cl%)*byh%/max-ya%:x1%=xc%+n*di%+incr%+xa%
3440  linef x%,y%,x1%,y1%
3450  pcircle x%,y%,largh%+3:if ev$(n,cl%)="*" then pcircle x1%,y1%,largh+10
3460  x%=x1%:y%=y1%:next
3470  pcircle x%,y%,largh%+3:i%=0:j%=0:gosub ESTREMI:i%=1:gosub SETWIDTH
3480  testo$=nome$(cl%):x%=xa%+(xb%-xa%)/2-8*len(testo$)/2:y%=yb%-4
3490  if ncl%<>1 then x%=xa%+((xb%-xa%)/ncl%+10)*j
3500  if fl%=2 then x%=xa%+((xb%-xa%)/(ncl%+1)+10)*j
3510  pcircle x%+xa%,y%-ya%-3,5:x%=x%+10:gosub TEXT
3520  return
3530  ISTOMULT:
3540  rem         ________________________
3550  rem         | istogrammi multipli |
3560  rem         ------------------------
3570  if cu%=0 then ?"non ho dati in memoria":return
3580  gosub ALLW2:gotoxy 0,0:?"---> ISTOGRAMMA MULTIPLO:":?:?
3590  input "Affiancati o Sovrapposti (a/s)";r$
3600  if r$="s" or r$="S" then goto SOVRAPPOSTI
3610  gosub MULTIGEN:if ncl%>cu% then ncl%=1:return
3620  input"distanza tra i gruppi di barre (in pixels)";dist%:col%=0
3630  poke systab+24,1
3640  largh%=(bxw%-(elem%(c%(0))*dist%))/(elem%(c%(0))*ncl%)
3650  di%=largh%*ncl%+dist%:incr%=largh%*ncl%/2
3660  gosub AXES:dist%=largh%*(ncl%-1)+dist%
3670  ISTOMULTSUB:
3680  poke systab+24,1
3690  xc%=xc%-largh%:for j=0 to ncl%-1
3700  cl%=c%(j):xc%=xc%+largh%:col%=col%+1
3710  gosub SUBBARRE:next
3720  poke systab+24,0
3730  if fl%=2 then return
3740  erase c%:ncl%=1:a%=inp(2):return
3750  CARTEMULT:
3760  rem         ________________________
3770  rem         | cartesiani multipli |
3780  rem         ------------------------
3790  if cu%=0 then ?"non ho dati in memoria":return
3800  gosub ALLW2:gotoxy 0,0:?"---> CARTESIANI MULTIPLI:":?:?
3810  gosub MULTIGEN:if ncl%>cu% then ncl%=1:return
3820  di%=bxw%/elem%(c%(0)):col%=0:incr%=5
3830  input"larghezza del tratto (1-20)";largh%
3840  poke systab+24,1
3850  gosub AXES:incr%=5
3860  for j=0 to ncl%-1:cl%=c%(j):col%=col%+1
3870  gosub SUBCARTE:next:erase c%:ncl%=1
3880  poke systab+24,0
3890  a%=inp(2):return
3900  MULTIGEN:
3910  rem         ______________________________________________________
3920  rem         | sub. generica calcolo dati per tracciamenti multipli |
3930  rem         ------------------------------------------------------
3940  input"quante classi vuoi rappresentare ";ncl%:?:?"classi usate:"
3950  for n=0 to cu%-1:?n+1;":";nome$(n);" ";:next:?:?
3960  if ncl%>cu% then ?"non esistono ";ncl%;" classi in memoria":return
3970  option base 0:dim c%(ncl%-1)
3980  if fl%=2 then ?"-Il dato '1' sara` rappresentato con un diagramma ";x$
3990  for n=0 to ncl%-1
4000  ?n+1;":";:gosub ASKCLASS:if cl%=-1 then goto 4000
4010  c%(n)=cl%:next
4020  rem >>>> trova classe max <<<<
4030  max=0:mx=0:tp%=0
4040  for n=0 to ncl%-1:for j=0 to elem%(n)-1
4050  if va(j,c%(n))>max then max=va(j,c%(n))
4060  next
4070  if max>mx then mx=max:tp%=n
4080  next
4090  swap c%(tp%),c%(0)
4100  cl%=c%(0):gosub AXESDAT
4110  swap c%(tp%),c%(0)
4120  return
4130  SOVRAPPOST1:
4140  rem           ________________________
4150  rem           | istogrammi sovrapposti |
4160  rem           ------------------------
4170  input"quante classi vuoi utilizzare";ncl%
4180  if ncl%>cu% then ?"non sono state definite abbastanza classi":return
4190  option base 0:dim c%(ncl%-1)
4200  ?:?"classi usate:":for n=0 to cu%-1:?n+1;":";nome$(n);" ";:next:?:?
4210  for n=0 to ncl%-1:?n+1;":";:gosub ASKCLASS:if cl%=-1 then goto 4210
4220  c%(n)=cl%:next
4230  rem >>>>>>>> MATRICE SOMMA <<<<<<<<<<
4240  max=0:for n=0 to ncl%-1:if elem%(c%(n))>max then max=elem%(c%(n))
4250  next:elem%(cpu%)=max:for n=0 to max-1:va(n,cpu%)=0:next
4260  for n=0 to ncl%-1:for j=0 to max
4270  va(j,cpu%)=va(j,cpu%)+va(j,c%(n)):next:next
4280  for n=0 to max-1:com$(n,cpu%)=com$(n,c%(0)):next
4290  cl%=cpu%:gosub AXESDAT
4300  input"distanza barre ";dist%
4310  for n=0 to elem%(cpu%)-1:va(n,cpu%)=0:next
4320  largh%=(bxw%-(elem%(cpu%)*dist%))/elem%(cpu%)
4330  di%=largh%+dist%:incr%=largh%/2:gosub AXES
4340  rem >>>>>>>> TRACCIATURA <<<<<<<<<<
```

```
4350 poke systab+24,1
4360 for j=0 to ncl%-1:color 1,1,1,j+1,2
4370 for n=0 to elem%(c%(j))-1
4380 y%=yc%-(va(n,c%(j))+va(n,cpu%))*byh%/max-5:x%=xc%+n*(largh%+dist%)+5
4390 i%=x%+largh%:j%=yc%-va(n,cpu%)*byh%/max-5:gosub VBAR
4400 va(n,cpu%)=va(n,cpu%)+va(n,c%(j)):next
4410 testo$=nome$(c%(j)):y%=yb%-16:x%=xa%+((xb%-xa)/ncl%+10)*j
4420 i%=x%+12:j%=y%+12:gosub VBAR:y%=j%:x%=i%+2:gosub TEXT
4430 next:erase c%:ncl%=1
4440 poke systab+24,0
4450 a%=inp(2):return
4460 AREE:
4470 rem         ______________________
4480 rem         | diagrammi ad aree |
4490 rem         ----------------------
4500 gosub ALLW2:gotoxy 0,0:?"--> Diagrammi ad aree:"
4510 gosub MULTIGEN:if ncl%>cu% then ncl%=0:return
4520 poke systab+24,1
4530 di%=bxw%/elem%(c%(0)):incr%=5:gosub AXES
4540 for j=0 to ncl%-1
4550 cl%=c%(j):gosub AREESUB
4560 next:poke systab+24,0
4570 ncl%=1:erase c%
4580 a%=inp(2):return
4590 AREESUB:
4600 rem        _________________________
4610 rem        | traccia la classe cl% |
4620 rem        -------------------------
4630 for n=0 to elem%(cl%)-2
4640 x%=n*di%+xc%+xa%+incr%:x1%=(n+1)*di%+xc%+xa%+incr%
4650 y%=yc%-va(n,cl%)*byh%/max-ya%:y1%=yc%-ya%-va(n+1,cl%)*byh%/max
4660 a%=y%-y1%:b%=x1%-x%:tp%=0:if (n mod 2)=0 then tp%=1
4670 for k=tp% to b% step j+2
4680 linef x%+k,yc%-ya%,x%+k,y%-a%*k/b%
4690 next
4700 next
4710 testo$=nome$(cl%):x%=xa%+((xb%-xa%)/ncl%)*j:y%=yb%-ya%-16
4720 if fl%=2 then x%=xa%+((xb%-xa%)/(ncl%+1))*ncl%
4730 for k=x% to x%+12 step j+2:linef k,y%,k,y%+12:next
4740 x%=x%+20-xa%:y%=y%+ya%+10:gosub TEXT
4750 return
4760 MISTO1:
4770 rem        ____________________
4780 rem        | aree+istogramma |
4790 rem        --------------------
4800 gosub ALLW2:gotoxy 0,0:?"---> Aree+Istogrammi:"
4810 fl%=2:x$="ad Area."
4820 gosub MULTIGEN:if ncl%>cu% then return
4830 input "Distanza tra i gruppi di barre";dist%
4840 poke systab+24,1
4850 largh%=(bxw%-(elem%(c%(0))*dist%))/(elem%(c%(0))*(ncl%-1))
4860 incr%=largh%*(ncl%-1)/2:di%=largh%*(ncl%-1)+dist%
4870 gosub AXES:dist%=largh%*(ncl%-1)+dist%
4880 cl%=c%(0):j=1:gosub AREESUB:ncl%=ncl%-1
4890 for n=0 to ncl%-1:c%(n)=c%(n+1):next
4900 col%=0:gosub ISTOMULTSUB
4910 fl%=0:poke systab+24,0:erase c%:ncl%=1
4920 a%=inp(2):return
4930 MISTO2:
4940 rem        _____________________
4950 rem        | carte+istogrammi |
4960 rem        ---------------------
4970 gosub ALLW2:gotoxy 0,0:?"--->Cartesiano+Istogrammi:"
4980 fl%=2:x$="Cartesiano."
4990 gosub MULTIGEN:if ncl%>cu% then return
5000 swap c%(0),c%(ncl%-1)
5010 input "Distanza tra i gruppi di barre";dist%
5020 poke systab+24,1
5030 largh%=(bxw%-(elem%(c%(0))*dist%))/(elem%(c%(0))*(ncl%-1))
5040 incr%=largh%*(ncl%-1)/2:di%=largh%*(ncl%-1)+dist%
5050 gosub AXES:xs%=xc%:ys%=yc%
5060 ncl%=ncl%-1:col%=0:gosub ISTOMULTSUB:col%=col%+1
5070 xc%=xs%:yc%=ys%
5080 cl%=c%(ncl%):j=ncl%:largh%=3:gosub SUBCARTE
5090 erase c%:ncl%=1:fl%=0
5100 a%=inp(2):return
5110 TOOLS:
5120 rem      ___________________________
5130 rem      | T O O L S     M E N U |
5140 rem      ---------------------------
5150 gosub ALLW2:gotoxy 0,0:?" **** BUSIGRAF ****"
5160 ?:?"UTILITY MENU:":?
5170 ?"[1] SORT NUMERICO","[2] SORT NUMERICO INVERSO"
5180 ?"[3] SORT ALFABETICO","[4] SORT ALFABETICO INVERSO"
5190 ?"[5] OPERAZIONI I/O COL DISCO"
5200 ?"[6] OUTPUT MENU":?"[7] INPUT MENU"
5210 ?:?"classi utilizzate:":if cu%=0 then ?"-Nessuna-":goto 5230
5220 for n=0 to cu%-1:?n+1;":";nome$(n);" ";:next:?
5230 ?:?"cosa scegli ?";:gosub ASKNUM
5240 a%=a%-3
5250 if a%>-3 and a%<2 then a%=1
5260 if a%=3 then SEZOUT
5270 if a%=4 then SEZIN
5280 on a% gosub SORT,DISCO
5290 goto TOOLS
5300 SORT:
5310 rem     _________________
5320 rem     | ordinamenti |
5330 rem     -----------------
5340 gosub ASKCLASS:if cl%=-1 then return
```

lizzatala, il numero del dato da modificare al suo interno. Appare per la prima volta una chiamata alla routine STAMPAA. Il suo compito è minimo, ma verrà dopo.

1640-1720 (CANCMEM): cancella una classe dalla memoria. La cancellazione in realtà si riduce al decremento di una unità della variabile cu% che è il numero delle classi attualmente definite.

1730-1830 (NUMSTAMP): stampa sul video i dati numerici della classe indicata. Notare che oltre al principale NUMSTAMP, la routine ha un altro ingresso: NUMSTAMP1. Esso semplicemente salta l'input della linea 1780 (che guarda caso è una chiamata ad una subroutine, ASKCLASS) ed è usato dalla routine...

1840-1980 (RIEMPIMENTO): eccola finalmente. Essa svolge (lo si era già capito) il compito di riempire la classe cl% a partire dal dato da%, stampando anche tutti i valori precedentemente definiti.

1990-2050 (STAMPAA): ecco un'altra vecchia conoscenza. Come detto prima stampa il dato a% della classe cl% nella sua posizione nella lista. Notare il ricorso allo statement PRINT USING per incolonnare i valori. A proposito: avevate notato che mettendo in un programma PRINT USING "#.#"; N dove N=0, si ottiene −0 e che PRINT USING "## / xxxxx / ##.#"; n; x$; a (che dovrebbe stampare un numero, una stringa di cinque caratteri e un altro numero) semplicemente non funziona?

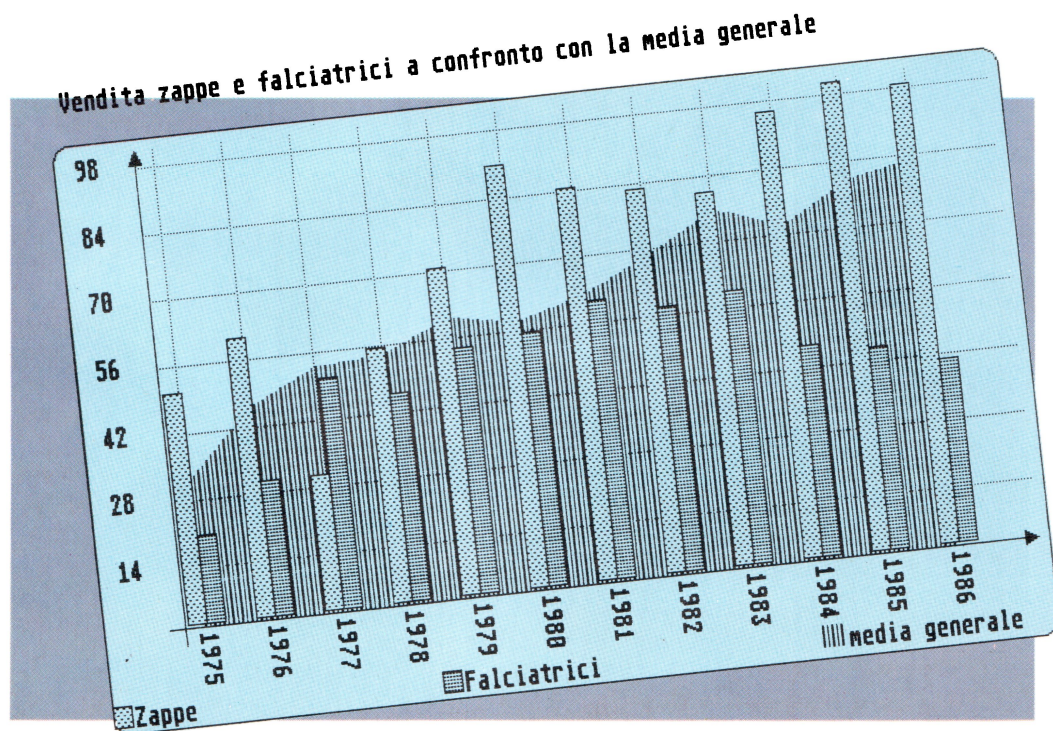
2060-2140 (ASKCLASS): altra routine usatissima; chiede il numero o il nome della classe da utilizzare, il cui numero torna in Cl%. Se il valore inserito è fuori dal range delle classi definite torna cl%=−1.

2150-2200 (ASKNUM): attende che venga inserito un numero dalla tastiera, senza aspettare la pressione di [RETURN]. Viene utilizzata l'istruzione "a%=INP(2)", che attende la pressione di un tasto e pone in a% il codice ASCII del tasto premuto.

2210-2420 (SEZOUT): eccoci al secondo menu operativo, l'OUTPUT MENU, che gestisce l'accesso alle routine grafiche. Notare che nel caso il tasto premuto sia [3], istogramma orizzontale, viene posta r$="or", avvertendo così la routine degli istogrammi, che è una sola, che il grafico va sviluppato in orizzontale.

2450-2690 (AREOGRAMMI): prima routine grafica, corrispondente all'opzione [1] dell'output menu. Viene liberato sul video lo spazio necessario e contornato da un box a spigoli arrotondati (2510) e sono calcolate le coordinate del centro della circonferenza (2520) in modo tale che sia al centro della metà di sinistra della output window. Si occupa della tracciatura il loop 2550-2590. Al suo interno la variabile 'an1' assume il valore in centesimi di grado dell'angolo corri-

spondente al dato desiderato e 'an', sempre in centesimi di grado, l'angolo già tracciato .'an' viene così incrementata ad ogni giro del valore 'an1' (2590), che prenderà il valore corrispondente al dato successivo; la linea 2570 si occupa di spostare il centro dello spicchio se il dato va evidenziato.
Si occupa della tracciatura effettiva la potente istruzione PCIRCLE (2580). Sempre all'interno del loop si stampa la legenda. Viene tracciato un quadratino campito come lo spicchio corrispondente e a fianco il nome del dato, con la subroutine TEXT, che vedremo alla fine.
Si noti che durante questa e tutte le altre routine grafiche si esegue POKE systab+24, 1 che accelera l'esecuzione delle istruzioni grafiche, disabilitando però ogni INPUT da tastiera, che viene riabilitato con POKE systab+24,0.
2700-2960 (BARRE): questa routine ha due ingressi: questo, e quello alla linea 2820, SUBBARRE, che in effetti è il più usato. Questo perchè dalla linea 2700 parte una routine di inserimento dati che viene usata solo per tracciare istogrammi proporzione che dal valore del dato ricava l'altezza delle barre è alla linea 2860, seguita dalla sua equivalente per gli istogrammi orizzontali (2870 if r$="or"...). Anche in questo caso sono stampati il quadratino con la retinatura e il nome della classe corrispondente. Le coordinate di stampa vengono calcolate alle linee 2900-2920 insieme alla stringa con il nome della classe.
2970-3090 (AXESDAT): ecco una routine chiamata molto spesso nella produzione dei grafici: essa, a partire dal numero della classe in cl% e dalle coordinate della finestra di visualizzazione (xa%, ya% e xb%, yb%), calcola il centro degli assi e le loro rispettive lunghezze (bxw% e byh%), badando di lasciare lo spazio vicino ad entrambi per i valori numerici e i nomi dei valori.
3100-3330 (AXES): ed ecco la routine complementare ad AXESDAT: essa disegna gli assi e i commenti, usando i dati generati da AXESDAT, insieme ad altri che vanno aggiunti da parte del programma che le gestisce. Ovvero di%, distanza tra una linea e l'altra, e incr%, distanza gli assi definiti da AXES e AXESDAT, disegnando agli estremi di ogni segmento un cerchio campito con la retinatura col%. Se si incontra un dato da evidenziare il cerchio campito è più grande. Notare che la misura del raggio dei cerchi non è fissa, ma cambia con il variare della larghezza scelta per il tratto (in largh%).
3530-3750 (ISTOMULT): ecco la routine che gestisce il tracciamento di istogrammi multipli, disegnati effettivamente da SUBBARRE, come abbiamo visto prima. Questa routine si occupa soltanto degli istogrammi affiancati, mentre per tracciare istogrammi sovrapposti il controllo viene passato a SOVRAPPOSTI, che più che una subroutine è un'estensione della routine ISTOMULT, in quanto è chiamata con GOTO. Anche ISTOMULT ha due ingressi: il secondo è ISTOMULTSUB, che è usato dalle due routine di grafici misti. ISTOMULTSUB traccia infatti con istogrammi affiancati le classi indicate nell'array c%(n) dove 'n' è il numero delle classi da rappresentare. Questo array viene definito all'in-

Volendo fare dei confronti a vari livelli tra dati non proprio omogenei (valori effettivi e una media) si può sempre differenziare la rappresentazione. Oltre al diagramma ad area, insieme agli istogrammi, si può rappresentare un diagramma cartesiano.

orizzontali. Il cuore del tracciamento istogrammi è invece 2820-2960, che rappresenta con una serie di barre i dati della classe indicata da cl%, considerando come origine degli assi il punto xc%, yc%, la larghezza delle barre, la variabile largh% e il colore della campitura col%. Tutte queste variabili sono passate dalla routine ISTOMULT, che si occupa della gestione di tracciamento di istogrammi multipli. I vari box sono tracciati dalla routine VBAR, in coda al programma come la già citata TEXT. La semplice della prima linea dall'asse Y. Questi dati cambiano a seconda delle necessità di ogni specifico grafico, impedendo di fondere AXES e AXESDAT in un'unica routine. Le linee 3150 e 3160 disegnano gli assi; i loops FOR/NEXT delle linee 3200-3250 e 3270-3220 disegnano una griglia di riferimento e i commenti, in orizzontale sull'asse Y e in verticale sull'asse X.
3340-3520 (SUBCARTE): routine di servizio nel tracciamento di diagrammi cartesiani. Rappresenta la classe in cl% neterno di MULTIGEN, chiamata alla 3610, e viene letto col loop 3690-3710. Alla fine l'array viene cancellato con ERASE c%. Per quanto non sia certo molto elegante, la ridefinizione degli array è molto comoda quando questi variano di dimensioni, come nel nostro caso, di volta in volta.
3750-3890 (CARTEMULT): la seconda routine per tracciamenti multipli è CARTEMULT. Come la precedente fa uso di MULTIGEN per l'input, mentre per la tracciatura si serve di SUBCARTE, vista

```
5350  rem >>>> routine sort generica <<<<<<
5360  cmb%=0
5370  for n=0 to elem%(cl%)-2
5380  on val(r$) gosub STNUM,STINUM,STALPH,STIALPH
5390  next:if cmb%=0 then return
5400  goto 5360
5410  STNUM:
5420  if va(n,cl%)>va(n+1,cl%) then gosub INVERT
5430  return
5440  STINUM:
5450  if va(n,cl%)<va(n+1,cl%) then gosub INVERT
5460  return
5470  STALPH:
5480  if com$(n,cl%)>com$(n+1,cl%) then gosub INVERT
5490  return
5500  STIALPH:
5510  if com$(n,cl%)<com$(n+1,cl%) then gosub INVERT
5520  return
5530  INVERT:
5540  swap va(n,cl%),va(n+1,cl%):swap com$(n,cl%),com$(n+1,cl%):cmb%=1
5550  swap ev$(n,cl%),ev$(n+1,cl%)
5560  return
5570  DISCO:
5580  rem          ________________________
5590  rem          | operazioni I/O DISK |
5600  rem          ------------------------
5610  gosub ALLW2:gotoxy 0,0:?" **** BUSIGRAF ****":?:?"I/O DISK MENU:":?
5620  ?"[1]        MEMORIZZA CLASSE":?"[2]        CARICA CLASSE"
5630  ?"[3]        CANCELLA DATI DAL DISCO"
5640  ?"[4]        <-- RITORNA UTILITY MENU"
5650  ?:?"classi usate:":if cu%=0 then ?"-Nessuna-":goto 5670
5660  for n=0 to cu%-1:?n+1;":";nome$(n);" ";:next:?
5670  ?"cosa scegli ?";:gosub ASKNUM
5680  on a% gosub SALVA,CARICA,CANCELLA
5690  if a%=4 then return
5700  goto DISCO
5710  CANCELLA:
5720  rem          _________________
5730  rem          | cancella file |
5740  rem          -----------------
5750  ?"CANCELLA FILE:":input"nome file da cancellare ([RET]=exit)";n$
5760  n$=left$(n$,8)+".DAT":if n$=".DAT" then return
5770  kill n$:return
5780  SALVA:
5790  rem        ________________
5800  rem        | save classe |
5810  rem        ----------------
5820  ?"REGISTRAZIONE CLASSE:":gosub ASKCLASS:if cl%=-1 then return
5830  ?:input"Che nome assegno al file ";n$:if asc(n$)=13 then return
5840  n$=left$(n$,8)+".DAT":n=elem%(cl%)
5850  open "O",#1,n$:write#1,n
5860  for n=0 to elem%(cl%)-1:write#1,com$(n,cl%),va(n,cl%),ev$(n,cl%)
5870  next:close #1:return
5880  CARICA:
5890  rem        ________________
5900  rem        | load classe |
5910  rem        ----------------
5920  if cu%>classi then ?"non ho spazio in memoria per altre classi":return
5930  ?"CARICAMENTO CLASSE:":cl%=cu%:cu%=cu%+1
5940  ?:input"Nome file da caricare ([RET]=exit)";n$:n$=left$(n$,8)+".DAT"
5950  if n$=".DAT" then cu%=cu%-1:return
5960  open "I",#1,n$
5970  input#1,n:elem%(cl%)=n
5980  for n=0 to elem%(cl%)-1:input#1,com$(n,cl%),va(n,cl%),ev$(n,cl%)
5990  next:close #1
6000  input "che nome assegno alla classe caricata";nome$(cl%)
6010  return
6020  UPPERCASE:rem >>> n$ ---> MAIUSCOLO <<<
6030  a$="":for n=1 to len(n$):b$=mid$(n$,n,1)
6040  if asc(b$)>96 and asc(b$)<122 then a$=a$+chr$(asc(b$)-32) else a$=a$+b$
6050  next:n$=a$:return
6060 . rem        ____________________________________________
6070  rem        | E R R O R   T R A P P I N G   R O U T I N E |
6080  rem        --------------------------------------------
6090  if erl<5880 or erl>6010 then goto 6120
6100  if err=19 then close:?"Errore di caricamento. RIPETERE":?:?:resume 5940
6110  if err=53 then close:?"File inesistente.":cu%=cu%-1:resume 6010
6120  on error goto 0:?"linea ";erl
6130  stop
10000 end
10010 SETWIDTH:poke contrl,16:poke contrl+2,1:poke contrl+6,3
10020 poke ptsin,i%:poke ptsin+2,0:vdisys(1):return
10030 SETTYPE:poke contrl,15:poke contrl+2,0:poke contrl+6,1
10040 poke intin,i%:vdisys(1):return
10050 VBAR:poke contrl,11:poke contrl+2,2:poke contrl+6,0
10060 poke contrl+10,1:poke ptsin,x%:poke ptsin+2,y%
10070 poke ptsin+4,i%:poke ptsin+6,j%:vdisys(1):return
10080 VRBOX:poke contrl,11:poke contrl+2,2:poke contrl+6,0
10090 poke contrl+10,8:poke ptsin,x%:poke ptsin+2,y%
10100 poke ptsin+4,i%:poke ptsin+6,j%:vdisys(1):return
10110 TEXT:poke contrl,8:poke contrl+2,1:poke contrl+6,len(testo$)
10120 for j%=0 to len(testo$)-1:poke intin+2*j%,asc(mid$(testo$,j%+1,1)):next
10130 poke ptsin,x%:poke ptsin+2,y%:vdisys(1):return
10140 BASELINE:poke contrl,13:poke contrl+2,0:poke contrl+6,1
10150 poke intin,i%:vdisys(1):return
10160 ESTREMI:poke contrl,108:poke contrl+2,0:poke contrl+6,2
10170 poke intin,i%:poke intin+2,j%:vdisys(1):return
10180 LINDEF:poke contrl,113:poke contrl+2,0:poke contrl+6,1:poke intin,i%
10190 vdisys(1):poke contrl,15:poke contrl+2,0:poke contrl+6,1
10200 poke intin,7:vdisys(1):return
10210 ALLW2:color 1,0,1,1,1:x%=0:y%=0:i%=640:j%=440:gosub VBAR:return
```

in precedenza. Anche qui (3860-3870) si ha un loop di lettura dell'array c%(n), che viene ugualmente cancellato dopo l'uso.

3900-4140 (MULTIGEN): ed ecco la routine che si occupa degli input multipli.

È questa che definisce di volta in volta c%() alla 3970. C'è anche un'OPTION BASE=0, che potrebbe sembrare superflua. Però non avendola, ogni tanto capita che il BASIC perda il valore di c%(0). La linea 3980, basandosi sul flag fl% si accorge se MULTIGEN è stata chiamata da una routine di tracciamento misto (fl%=2) e in quel caso avverte che la classe inserita per prima sarà rappresentata diversamente. Ma il compito di MULTIGEN, più che riempire c%(), è trovare la classe che, tra quelle inserite, raggiunge il valore più alto (4020-4090) e porla in c%(0), cosicchè, posto cl%=c%(0) (4100) la routine AXESDAT lavori coi valori massimi ottenibili, evitando così che le routine di tracciatura vadano fuori schermo. Si notino gli SWAP c%(0), c%(tp%) delle 4090 e 4110 (tp% punta all'elemento di c%() che raggiunge il massimo). In questo modo alla 4090 in c%(0) va la classe col valore massimo, e, settato cl%, essa torna al suo posto, assicurando così che la tracciatura avvenga nell'ordine richiesto.

4130-4450 (SOVRAPPOSTI): questa è una delle routines più lunghe: è chiamata da ISTOMULT e stampa istogrammi sovrapposti. La sua lunghezza è causata dal dover compiere lei stessa i compiti di input e di tracciatura che negli altri casi sono svolti da subroutines particolari. Questo è dovuto al fatto che per gli istogrammi sovrapposti è necessario non solo conoscere le classi che dovranno essere rappresentate, ma anche definire exnovo una classe che contenga per ogni dato **la sommatoria** dei dati corrispondenti delle varie classi. MULTIGEN non ha questa possibilità, che viene così svolta dalle linee 4230-4330. La nuova matrice deve fare parte dell'array va(x, y), che contiene **tutti** i dati numerici inseriti, per essere riconosciuta dalle routines AXESDAT e AXES. Per questo all'inizio (linea 1050) esso viene definito con una dimensione eccedente le reali necessità (classi + 1). Lo spazio che si forma così è disponibile per SOVRAPPOSTI e viene identificato da AXESDAT con va(x, classi + 1).

Anche la tracciatura è autogestita, da 4340 a 4450, con due FOR/NEXT nidificati.

4460-4580 (AREE): dopo la routine di tracciatura più complessa ecco la più semplice. Essa infatti non deve occuparsi d'altro che di chiamare MULTIGEN e, col loop 4540-4560, AREESUB, che traccia il grafico ad area della classe cl%.

4590-4750 (AREESUB): traccia il grafi-

co ad area della classe cl%. La lunghezza delle linee fra un valore e l'altro è calcolata direttamente dalla 4680.
4760-4920 (MISTO1): gestisce la stampa di istogrammi sovrapposti a un diagramma ad aree. Si pone per questo il flag fl%=2 e x$="ad Aree", per comunicare a MULTIGEN di regolarsi come abbiamo visto prima. Definita c%() si passa a tracciare il grafico ad area con AREESUB, passando cl%=c%(0). Dopo aver fatto poi scalare di un posto gli elementi di c%(), così da avere in c%(0) quello che stava in c%(1) (4890), si chiama ISTOMULT che provvede al resto.
Alla fine il flag torna a 0.
4930-5100 (MISTO2): ultima routine gestita dall'output menu. Traccia degli istogrammi sovrastati da un diagramma cartesiano ed è simile alla precedente.
La differenza è che al ritorno da MULTIGEN il numero della classe che stava in c%(0) va in fondo all'array (5000), per poter disegnare per primi gli istogrammi. Le coordinate degli assi sono poste in xs% e ys% perchè ISTOMULTSUB, chiamata da 5060, altera gli originari valori xc%, yc%, che vengono ripristinati in 5070, per passare il controllo a SUBCARTE (5080).
5110-5290 (TOOLS): è fatta. Siamo arrivati all'UTILITY MENU, e la strada è ormai alla fine. Le funzioni che seguiranno sono infatti le più semplici del programma.
5300-5560 (SORT): è la routine che si occupa di tutti gli ordinamenti, essendo l'algoritmo uguale per tutti, a differenza dei riferimenti a va(x,y) e a com$(x,y), che sono gestiti dalle varie 'miniroutines' STNUM (ordinamento numerico), STINUM (ordinamento numerico inverso), STALPH (ordinamento alfabetico), STIALPH (alfabetico inverso).
5570-5700 (DISCO): menu di gestione del floppy disk. Tutte le operazioni avvengono sul floppy di default (normalmente A).
5710-5770 (CANCELLA): con KILL cancella il file dal nome n$, al quale è aggiunta l'estensione ".DAT"
5780-5870 (SALVA): apre alla linea 5850 un file chiamato n$+".DAT" e al suo interno scrive per prima cosa (con WRITE#, più indicata di PRINT#) elem%(cl%), e in seguito i dati, inserendo com$ prima il nome (cpm$(n,cl%)) e poi il valore (va(n,cl%)).
5880-6010 (CARICA): cerca sul disco il file di nome n$+".DAT", lo apre e assegna a elem%(cl%) il primo valore letto. Su questo numero sarà basato il loop 5980-5990 che legge tutti i dati. Alla classe così definita sarà poi dato un nome inserito da tastiera, non il nome del file.
6020-6050 (UPPERCASE): porta in maiuscolo la stringa n$.
10000-10210: ecco le famose routine grafiche che appaiono chiamate in tutto il programma. Esse usano le routine VDI che gestiscono la grafica del 520 e possono essere usate in qualsiasi programma. Vediamole velocemente.
SETWIDTH: regola la larghezza delle linee, inserita in i% (da 1 a 25).
SETTYPE: seleziona il pattern delle linee, posto in i% (da 1 a 6).
VBAR: disegna un rettangolo campito col colore e la retinatura correnti: x%, y%=coordinate del punto in alto a sinistra; i%, j%=coordinate punto in basso a destra.
VRBOX: come sopra, ma il rettangolo ha gli angoli arrotondati.
TEXT: stampa la stringa testo$ a partire dalle coordinate **grafiche** x%,y%.
BASELINE: regola l'inclinazione delle scritte secondo l'angolo in i% (0,900, 1800, 2700).
ESTREMI: seleziona la forma degli estremi delle linee; i% inizio linea, j% fine linea (0=estremi piatti, 1=frecce, 2=arrotondati).
LINDEF: definisce il pattern delle linee, inserito in i% (es i%=&haaaa).
ALLW2: disegna con VBAR un rettangolo grande tutto lo schermo con il colore dello sfondo. In pratica cancella gli accessori della Output Window e il suo contenuto.

COSTANTI

CLASSI: indica il numero delle classi che Busigraf può gestire meno uno. È definita alla linea 1030 e può essere tranquillamente mutata.
PI: pigreco, usata da AREOGRAMMI.
cpu%: usata da SOVRAPPOSTI, è sempre CLASSI+1.
fr$** e **fr1$: stringhe usate da PRINT USING per la stampa formattata dei dati.

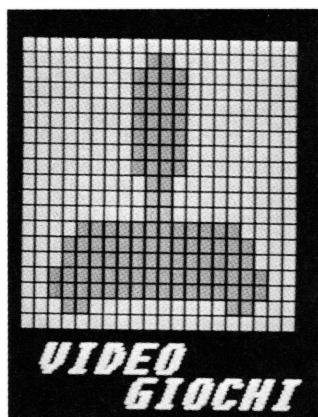
ARRAYS

va(x,y): valori numerici dei dati. X è il numero d'ordine (da 0 a 23) e Y è il numero della classe. Le dimensioni sono va (23, classi+1). La classe va (x, classi +1) contiene, durante SOVRAPPOSTI, in ogni elemento, la sommatoria degli elementi corrispondenti delle varie classi da rappresentare.
com$ (x,y): nomi legati ai valori X e Y come sopra; com$ (x, classi + 1) contiene in SOVRAPPOSTI, i nomi dei dati della classe numero c%(0).
ev$ (x,y): tabella delle evidenziature. Se ev$ (x,y) = "*" allora il dato X della classe Y va evidenziato.
nome$(n): nome della classe 'n'.
elem%(n): elementi usati da ogni classe. 'N' è il numero della classe.
c%(n): array definito da MULTIGEN, contiene normalmente ncl% valori, che corrispondono ai numeri delle classi da rappresentare nei tracciamenti multipli. Viene cancellato dopo ogni tracciamento e ridefinito all'occorrenza.

FLAGS

fl%: flag è normalmente 0, 1 se è in funzione la routine RIEMPIMENTO, 2 se è in funzione una delle due routines di tracciamento multiplo.
cmb%: flag usato da SORT. cmb%=0 indica che non sono stati fatti cambiamenti nell'ordine degli elementi, cmb%=1 il contrario.
r$: flag stringa con vari usi. R$ = "or" negli istogrammi orizzontali, in SORT R$ = "1"... "4" indica quale ordinamento effettuare dei quattro disponibili.

VARIABILI

cu%: numero delle classi usate.
ncl%: numero delle classi da rappresentare contemporaneamente. Definita da MULTIGEN, appare già a 1110 perchè è controllata anche da quelle routines che non la chiamano.
cl%: all'interno di **tutte** le routines, fa riferimento alla classe su cui si deve operare o si sta operando.
da%: usata da RIEMPIMENTO. Indica il numero del dato dal quale cominciare a chiedere i dati.
xa%, ya%: usata dalle routines grafiche. Sono coordinate **assolute** ovvero riferite non alla Output Window, ma all'angolo alto-sinistra del video.
xb%, yb%: come sopra. Sono le coordinate dell'angolo in basso a destra.
an e **an1:** usate da AREOGRAMMI. An% è l'angolo tracciato fino al momento in cui viene letta, an1% è l'angolo corrispondente al valore di va(n, cl%). Entrambe sono espresse in centesimi di grado sessagesimale.
xc%, yc%: in AREOGRAMMI sono le coordinate **relative alla OUTPUT WINDOW** del centro del diagramma; altrove sono le coordinate **assolute** dell'origine degli assi cartesiani, calcolate da AXESDAT.
bxw%, byh%: dimensioni degli assi cartesiani calcolate da AXESDAT.
max, mx e **max$:** usate da tutte quelle routines che devono calcolare i massimi delle classi e le stringhe più lunghe tra i nomi dei valori.
dist%: usata da BARRE e SUBBARRE. È la distanza tra una barra e la successiva in un istogramma.
largh%: larghezza delle barre negli istogrammi e del tratto nei diagrammi cartesiani.
di%: usata da AXES, è la distanza tra le linee verticali di un diagramma.
incr%: usata da AXES, è la distanza della prima linea verticale dall'asse Y.
col%: codice della retinatura corrente.
x%, y%, i%, j%: generalmente servono da input alle routines VDI.
a%, b%, tp%, s1%, s2%: variabili di comodo.
n, j, k: variabili di ciclo.

DI MAURO PAVONE

WINTER OLYMPICS

Categoria: Arcade
Produttore: Tynesoft
Configurazione:
Atari 130-800

Sul genere sportivo, come è noto, sono stati prodotti diversi giochi, in particolare, sugli sport "invernali" ne sono usciti molti; tra questi, è ben noto WINTER GAMES per la sua riuscita grafica, molto realistica, e per la sua giocabilità che, a volte, però, può risultare un po' complessa ed impegnativa. Bene! Per chi vuole cimentarsi in tutti quegli sport ambientali fra le nevi dei monti e i ghiacci delle piste, senza risultati disastrosi, in una sede divertente, curata graficamente, attraente sotto tutti gli aspetti, non ha da fare altro che recarsi alle OLIMPIADI INVERNALI. Per farlo bastano, naturalmente, il vostro buon vecchio Atari con il gioco WINTER OLIMPICS; così, avrete a vostra disposizione ben cinque diverse discipline sportive per scaricare le vostre energie di troppo.

– SPEED SKATING (pattinaggio di velocità) Per partire e continuare a pattinare in piena velocità dovete smanettare con il joystick nei due lati (come in buona parte dei giochi sportivi), finché non completerete il percorso consistente in ben 200 metri piani. Il tempo più basso vi aiuterà a raggiungere un punteggio più alto.

– SLALOM

In questa prova dovete scendere a valle evitando di urtare le bandierine o di passare all'esterno di esse, altrimenti vi verrà assegnata una penalità che peserà sul vostro tempo, ed evitando di urtare gli alberi, con conseguente passaggio immediato alla prova successiva. Per muoversi orizzontalmente spostate la leva a destra e a sinistra, lo sciatore si sposterà diagonalmente nella direzione scelta e tornerà in posizione verticale solo uando muoverete la leva nel senso opposto a quello corrente. Acquistare padronanza con questo tipo di movimento all'inizio può risultare difficoltoso, ma dopo un po' di pratica vi troverete sufficientemente abili per affrontare la discesa e imboccare lo spazio fra le bandierine che vi sarà segnalato con un suono.

– SKY-JUMP

Ecco una prova impegnativa fra le tante abbastanza semplici! Per riuscire a saltare dovete attentamente seguire le istruzioni che vi indicherò qui di seguito:

– muovere rapidamente la leva nei due lati in modo da fare aumentare la velocià che sarà importante per affrontare il salto (movimento analogo al pattinaggio)

– quando la discesa termina premete istantaneamente il bottone del joystick e spingete la leva in avanti, riuscendo così a saltare se tutto è stato fatto correttamente vedrete il vostro sciatore spiccare un salto proporzionato allo slancio iniziale e poco dopo l'atterraggio un valore numerico posizionato in alto a sinistra vi segnalerà la distanza raggiunta.

– DOWNHILL (discesa libera)

Il metodo è simile a quello della seconda prova, in questo caso, però, bisogna solo stare attenti a non urtare gli alberi.

– CRESTA RUN

Fra i cinque sport, la corsa su bob, si prospetta come la più divertente; ma bisogna fare attenzione perché anche se le regole da seguire sono poche bisogna saperle rispettare per superare "gloriosamente" la specialità.

In questa schermata viene raffigurata, in grande, a destra la pianta della pista (blu), in mezzo agli alberi, che si accorcerà man mano che la percorrerete con il vostro bob.

A sinistra c'è il riquadro che raffigura il vostro bob visto posteriormente, dentro il canale della pista, ed il percorso che c'è davanti a voi. Per curvare spostate la leva del joystick a destra e a sinistra, cercando di controllare il vostro mezzo senza farvi trasportare all'esterno della pista dalla forza centrifuga, contrapponendo il vostro movimento alla direzione della curva. Se riuscirete ad imboccare bene le curve prenderete l'adeguata velocità per fare un buon tempo. Vi accorgerete del termine del percorso quando vedrete sparire completamente la pista dal lato destro dello schermo.

Infine, vi ricordo che ad ogni prova potete scegliere se far pratica o gareggiare regolarmente contribuendo all'incremento del punteggio. Se sceglierete di gareggiare avrete per ogni specialità ben tre possibilità complete.

Grafica	7
Difficoltà	8
Velocità	6
Originalità	6
Suono	7

DI MAURO PAVONE

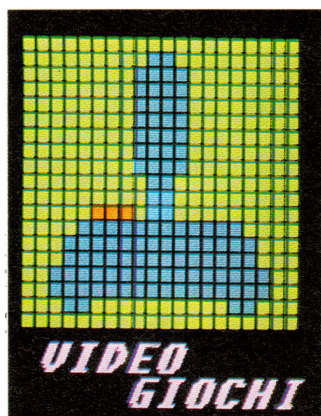
ALTERNATE REALITY

Categoria: Arcade
Produttore: Datasoft
Configurazione:
Atari ST + monitor a colori, Atari 130/800

Dopo pochi secondi di caricamento, sul video appare un'ampia visuale di una metropoli. Da quel momento inizia la vicenda che sarà alla base di tutto il gioco (e che potrai interrompere per passare al gioco vero e proprio senza perdere tempo).
Il traffico cittadino che disturba la quiete, verrà presto interrotto dall'arrivo di un'astronave che ti rapirà per portarti a XEBEC'S DEMISE, nella galassia più lontana.

Ottima la presentazione di questo singolare adventure che può sembrare l'inizio trionfale di un film fantascientifico, integrata, soprattutto, da una colonna sonora stupenda (da cantare, se volete, seguendo i testi 'dettati' dal video).
Per giocare, dopo aver formattato un disco, con l'opzione 'I', indispensabile per la memorizzazione del personaggio e dopo aver scelto se usare un carattere nuovo, uno preesistente o uno temporaneo, passerai alla scelta delle tue capacità iniziali. Nella parte superiore del video appaiono 6 caselle con dei numeri che variano ripetutamente, tu schiacciando RETURN fisserai tali valori che corrispondono alla resistenza, al fascino, alla forza, all'abilità, all'intelligenza e alla saggezza. Ognuna di queste qualità sarà importante per superare le diverse situazioni, più o meno impegnative, che si verificheranno nel corso del gioco, ma non sarà l'unica per fuggire della città.
Il paesaggio di XEBEC'S DEMISE è composto da alberghi utili per l'alloggio e le informazioni sul tempo e sulle date, le osterie che offrono intrattenimenti vari (anche molto utili e indicativi per arrivare alla soluzione del gioco), le banche, i negozi che ti possono far diventare ricco e quindi abbellire la tua persona ed, infine, i passaggi segreti presenti nei muri che sono visibili solo da un lato e attraversabili da quello invisibile (!). I personaggi che incontri possono essere fabbri, guaritori e forma di vita come gli umanoidi, i cittadini, i commercianti, i corrieri, le guardie, etc. In ogni caso, qualsiasi tipo d'incontro può essere rischioso data la tua inferiorità rispetto agli altri; comunque, essendo una cosa che si verficherà, devi imparare ad essere all'altezza delle straordinarie creature che incontrerai.

Per la tua sopravvivenza sono necessarie anche le armi che puoi trovare dopo un incontro, dopo aver scoperto un tesoro o comprarle da un fabbro, le posizioni possono invece essere svantaggiose, benigne, vantaggiose o pericolose ed, infine, il veleno è assolutamente da evitare per conservare la salute.
Per muoverti, puoi usare il joystick o la tastiera (tasti I-alto, K-basso, J-sinistra, L-destra). La ricerca della tua strada, nella città, è semplificata dall'uso di una bussola che puoi comprare premendo SELECT fino a che la vedi apparire sullo schermo; la tua direzione sarà sempre la più alta sullo schermo.
Per concludere la panoramica su questo splendido adventure, bisogna citare la finestra grafica (al centro del video) che offre una visuale sul paesaggio veramente ben realizzata, d'altra parte come tutto il programma.

Grafica	10
Difficoltà	8
Velocità	7
Originalità	8
Suono	9

DI MAURO PAVONE

BEACH HEAD II

The dictator strikes bach

Categoria: Arcade
Produttore:
Access Software
Distributore: Giucar
Configurazione:
Atari 130-800

Il dittatore colpisce ancora! Dopo il successo di Beach Head, ecco la seconda versione del famoso gioco che ci ha proposto come motivo di svago una "divertente" simulazione di battaglia, qui in veste rinnovata secondo le esigenze sempre più raffinate degli amanti di giochi di questo genere.
I giocatori possono essere uno o due.
Nel primo caso voi sarete gli alleati e dovrete sfidare il dittatore (quale non si sa!!!), nel secondo caso vi spartirete a vostra scelta i due ruoli.
I livelli di gioco sono 3 e c'è la possibilità di fare un po' di pratica prima di passare alla sfida vera e propria.
Nell'ATTACCO voi controllate l'elicottero sulla parte superiore che deve lanciare i soldati, disponendoli dietro i 4 muri protettivi posteriori senza che il dittatore li uccida con il suo cannone. Nel lasciarli dovete posizionarvi ad un'altezza adeguata in modo che i soldati non atterino nella foresta. Quando li avrete messi tutti al riparo l'elicottero sparirà dalla scena e un cursore punterà ciclicamente i 4 muri. Premendo FIRE sceglierete il muro segnato dal cursore dal quale uscirà un soldato che si porterà in prima linea con il movimento della manopola. Quando tutti i soldati saranno avanzati dovrete farne preparare uno all'attacco premendo FIRE, poi ripremendo farete uscire il compagno che vi coprirà le spalle mentre voi andrete davanti al cannone per lanciare una bomba (sempre con il FIRE).
Fino all'esaurimento degli uomini dovrete attaccare il dittatore senza eccessive perdite di vite, cercando di totalizzare il massimo punteggio.
Nel SALVATAGGIO il cannone in primo piano sarà, questa volta, la vostra arma. Il vostro compito è di sparare contro tutti i possibili attentatori dell'omino che attraversa lo schermo (l'alleato), cioè il carrarmato che tenta di investirlo, il furgone e l'uomo appostato in cima al muro della caserma che gli spara.
Concluso il tentativo di salvataggio, devi fuggire con il tuo elicottero attraversando un campo carico di insidie. Torrette e altre realizzazioni belliche tenteranno di distruggerti. Il carroarmato, potendosi muovere orizzontalmente, è molto più pericoloso, quindi ti conviene colpirlo immediatamente disponendoti alla sua altezza.
Per concludere trionfalmente la sfida, vi cimenterete in un duello con il dittatore stesso. In questa "manche" il sipario si apre su un fiume con due pontili ai lati dove ci saranno, il nemico a sinistra e il vostro uomo sul lato opposto. L'obbiettivo di questa BATTAGLIA è di uccidere l'avversario con 4 coltellate (ben piantate!) che voi lancerete premendo FIRE, indirizzando il lancio con la manopola del joystick. In questo simpatico "match", i due avversari si scambieranno messaggi ironici, frasi da fumétti e risate di vittoria, secondo lo spirito tanto avvincente di tutto il gioco.

Grafica	9
Difficoltà	8
Velocità	8
Originalità	9
Suono	7

PUNTI VENDITA ATARI 1987

CAMPANIA

2G - V. Dalmazia 53 - Salerno
ABS - V. Renzullo - Nola
AUTORADIO di Acanfora Rosa - Vico Ferrovia 11 - Napoli
BABY TOYS di Canetti SAS - V. Cisterna dell'Olio 5B - Napoli
CENTRO COMPUTER GRAFICA - P.tta Durante 7 - Napoli
CENTRO SANDOZ - P.zza Municipio 56 - Napoli
CERMA SAS - V. Giovanni Amendola 22/24 - Afragola
CF ELETT. PROFESSIONALE - C.so Vittorio Emanuele 54 - Napoli
CF ELETTRONICA - V. Luca Giordano 40/42 - Napoli
CF ELETTRONICA - V. G. Battisti - Napoli
CF ELETTRONICA PROFESS. - V. Marino 11/13 - Piano di Sore
COMMODORE CLUB CAMPANIA - V. Portalba 17/A - Napoli
COMPUTER CENTER SRL - P.zza S. Alfonso 19/A - Pagani
COMPUTER CLUB - V. Degli Orti 2 - Salerno
COMPUTER DAY SRL - V. Cilea 256 - Napoli
COMPUTER LAND SRL - V. Robertelli 17/B - Salerno
COMPUTER MARKET SRL - C.so Garibaldi 47 - Salerno
ELESYS di Carbone Anna - V. Mazzini 107 - Battipaglia
EUROMERCATO CAMPANIA SPA - V. Salvatore 1 - Casoria
GENERAL COMPUTER - C.so Garibaldi 56 - Salerno
GENERAL SYSTEM SRL - C.so Trieste 29 - Caserta
GLM COMPUTER SRL - C.so Garibaldi 141 - Palmi
GRUPPO BUSH SRL - Gall. Umberto I 55 - Napoli
INFORMATIC METHOD COMPUTER - V. A. Garofano 58 - Aversa
MUSICAL di Tutore Maria - V. S. Sebastiano 17 - Napoli
NEW OTTICA - Gall. Umberto I 55 - Napoli
ORION INFORMATICA - V. Virgilio 32 - Castellamare
PARIEL SRL - V. I Maggio 5 - Casoria
POLITECNICO ITALIANO - V. S. Arcangelo Abaiano - Napoli
STRUMENTI MUSICALI - V. S. Sebastiano 8 - Napoli
TECNOBYTE SAS - V. Risorgimento 53 - Marigliano
TOP ELECTRONICS - V. S. Anna dei Lombardi 16 - Napoli

EMILIA ROMAGNA

ARGNANI F.lli - P.zza Libertà 5 - Faenza
COMPUTER FACILE - V. Don Minzoni 4/B - Bologna
COMPUTER HOUSE di Sassi M. Pia - V. Secchi 28/D - Reggio Emilia
COMPUTERSHOP di Ronchi Tonino - V. Emilia 199/B - Imola
DIMENSIONE COMPUTER - V. E. De Amicis 16/A - Porto Maggio
EASY COMPUTER - V. Lagomaggio 50 - Rimini
GENIUS di Varani - V. Taverna 44/E - Piacenza
GRIFO SNC - V. Dante 1 - S. Giorgio D
HOME E PERSONAL COMPUTER - P.zza Melozzo 1 - Forlì
MICROINFORMATICA di Ruini e Mauro - P.zza M. Partigiani 31 - Sassuolo
OCA INFORMATICA - P.zza G. da Verazzano 6 - Bologna
ORSA MAGGIORE - P.zza Matteotti 20 - Modena
PAOLINI MARIO - V. Candiano 23 - Ravenna
S.C. COMPUTER di G. Capra e C. SNC - V. S. Martino 4 - Castel. S.P.
SOFT & COMPUTER - V. Carlo Mayr 85 - Ferrara
TECNO CONSULTING - V. Catalani 3 - Parma
TRIA ELETTRONICA SRL - V. Zacconi 28/A - Parma
TUTTO PER IL BIMBO di Babini Flavio - V. G. Rignoli 15 - Forlì

LAZIO

ABBEY ROAD SNC - V. Siria 5/7 - Roma
ADM SRL - V. Tacito 88 - Roma
ALFA LEASING SRL - V. Illiria 18 - Roma
ALL COMPUTER SRL - V. Catalani 31 - Roma
APC SRL - V. Catalani 19/23 - Roma
BANDIERA SRL - V. Cavour 125 - Roma
BIMBICA ANTONIO - V. Tagliamento 57 - Roma
CHERUBINI SNC - V. Tiburtina 360 - Roma
CIAMPI SRL - V. Vespasiano 34 - Roma
COMPUTEL - V. E. Rolli 33 - Roma
COMPUTER - V. E. Rolli 3 - Roma
COMPUTER FRIEND SRL - V. A. Romano 3 - Roma
COMPUTERLINE SRL - V. M.A. Colonna 12 - Roma
COMPUTIME - V. Cola di Rienzo 28 - Roma
COMPUTIME - V.le Parioli 25 - Roma
COMPUTRON SHOP - Largo Fornao 7/B - Roma
D'ALBORE CRISTINA - V. Principe Amedeo 52 - Roma
DIGITRON SRL - V. Lucio Elio Seiano 15 - Roma
DISITACO SRL - V. Poggio Moiano 34/C - Roma
DUE EMME ELETTRONICA SRL - V. Britannia 17 - Roma
ELETTRONICA 2003 - V. Antonio Cozzi 13 - Roma
MASTERBIT SNC - V. Dei Romagnoli 35 - Ostia
MIDI WARE SRL - V.le Parioli 101/C - Roma
ORGANIZ. SERVIZ. ELETTRONI - V. Tuscolana 465 - Roma
SICOM ITALIA SPA - P.zza Regina Margherita - Roma
SYNCROM SRL - V. F. Massimo 32 - Roma

LIGURIA

2002 ELETTROMARKET - V. Monti 15/R - Savona
ABM COMPUTERS SRL - P.zza De Ferrari 24/R - Genova
ALFASOFT SAS - V. D.G. Storace 4/R - Sampierdarena
COMPUTER LIFE SNC - V. Trento Trieste 1 - Ventimiglia
COOP. LIBRARIA UNIVERSITARIA - Salita Inferiore della Noce 10/R - Genova
GAGGERO LUIGI - P.zza Cinque Lampadari 63/65R - Genova
PAGLIALUNGA S.D.T. - V. Mazzini 4 - Rapallo
R & R ELETTRONICS - V. F.lli Canepa 94 - Serrà Ricco
SCK COMPUTER SNC - V. Piave 78/R - Savona
UN.EL.CO di Crespi e Corte - V. Roma 146 - Sanremo

LOMBARDIA

2 M ELETTRONICA SRL - V. Sacco 3 - Como
ABC INTERNATIONA SRL - V. C. Battisti 21 - Albiate Brianza
A.I.S. INTERNATIONAL SRL - V. Madonnina 33 - Agrate Brianza
ANTICA CASA MUSICAL SNC - V. G. Verdi 31 - Bergamo
BERNASCONI MARIO & C. - V. A. Saffi 88 - Varese
BIT 84 SAS - V. Italia 4 - Monza
BOSONI - P.zza Tricolore 2 - Milano
COMPUTER & C. SNC - P.zza Indipendenza 4 - Seregno
COMPUTER HOUSE - V. Maffei 11/A - Sondrio
COMPUTER LINE SRL - V. Maroncelli 12 - Milano
COMPUTER SHOP - V. A. da Brescia 2 - Gallarate
COMPUTER SHOP - V. Vittorio 9 - Capriate S.
COMPUTER STUDIO SNC - V. Einaudi 13 - S. Antonio P
COMPUTER & GRAPHICS - V. Antica Regina 173 - Domaso
DELTRON SRL - V.le Gran Sasso 50 - Milano
DISCOUNT MUSIC CENTER SRL - V.le Monza 16 - Milano
DOZIO SYSTEM - V. Marco D'Oggiono 11/A - Lecco
EDELKTRON - P.zza Pattari 2 - Milano
EDS COMPUTERS SRL - C.so Porta Ticinese 4 - Milano
ELETTRONICA INDUSTRIALE - V. S. Pellico - Villongo
ELETTRONICA SESTESE - V. Boccaccio 178/180 - Sesto S.G.
EMI COMPUTER - V. Azzone Visconti 39 - Monza
EVERY F.H. SRL - V. Vitruvio 3 - Milano
GBC ITALIANA SPA - V.le Matteotti 66 - Cinisello B.
GIGLIONI SRL - V.le Don Sturzo 45 - Milano
HEX ELECTRONICS SAS - V.le Jenner 16 - Milano
IL DATO di Noris Maria - V. Provinciale 66/E - Albino
IL PAPIRO - V. Trento 18 - Toscolano M.
INFORMATICA 2000 SRL - V.le Stazione 16/C - Brescia
INFORMATICA SERVICE - V. Negroli 26/2 - Milano
L'AMICO DEL COMPUTER - V. Castellini 25 - Melegnano
LECCOLIBRI LIBRERIA FUMAGALLI - V. Cairoli 48 - Lecco
LOGICAL STATION 3001 SRL - V. delle Asole 2 - Milano
MANTOVANI TRONIC'S - V. Caio Plinio 11 - Como
MARCUCCI SPA - V. F.lli Bronzetti 3/A - Milano
MICROTHERMIK SAS - V.le Rimembranze 93 - Sesto S.G.
MONITOR ELECTRONICS - V. De La Salle 10 - Milano
MOUSE SRL - V. Volta 11 - Seregno
MULTISYSTEM SAS - V. Aurora 6 - Cinisello B.
MUSIC POOL SOC. COOP. SRL - V. Archimede 22 - Milano
MUSIC TECHNOLOGY SRL - V. C. Colombi 12 - Milano
NEBEL ELECTRONICS SRL - V. V. Emanuele 65 - Vimercate
NEW GAME SNC - C.so Garibaldi 199 - Legnano
NUOVA NEWEL ELETTRONICA SAS - V. Mac Mahon 75 - Milano
POLLI SRL - V. Martiri Libertà 103 - Lissone
PRISMA SNC - V. Ghisleri 55 - Cremona
REPORTER SNC - C.so Garibaldi 25 - Cremona
RIVOLA SNC - V. Vitruvio 43 - Milano
SENNA G. FRANCO & C. SNC - V. Calchi 5 - Pavia
SIGMA SAS - V. Canelli 25 - Milano
SONDRIO COMPUTER SAS - V. Mazzini 44 - Sondrio
SUPER GAMES SAS - V. Vitruvio 38 - Milano
TECNOTRON di Iannucci - V. Breda 274 - Fara Gera D.
TINTORI - V. Brosetta 1 - Bergamo

PIEMONTE

CASA MUSICALE SCA - V. Ormea 66 - Torino
COMPUTER SHOP SAS - V. Nizza 9 - Torino
COMPUTING SRL - P.zza Risorgimento - Vercelli
EMMESOFT - V. A. Alberti - Torino
RECORD - C.so Alfieri 1 - Asti
ROSSI COMPUTERS SNC - C.so Nizza 42 - Cuneo
SUONO - V. Po 40 - Torino
TEOREMA SRL - V.le Losanna - Biella

PUGLIA

AMORE COMPUTER - V. Cavalieri di Malta 59/61 - Putignano
ARTEL - V. Fanelli 206/16 - Bari
ARTEL - V. Palete 3/7 - Modugno
BIBO SYSTEM - P.zza S.'Angelo 13 - Manduria
CARTOLIBRERIA RIZ - V.le Luigi Sturzo 49 - Bari
COMPUTER'S ARTS I - V. Regina Elena 101 - Taranto
DISCORAMA SRL - C.so Cavour 99 - Bari
EDZ di Nicola Zuc - V. Umberto I 20 - Pulsano
ELECTRONIC SYSTEM - V. Nizza 21/25 - Castellana
ELIOSTATIK - V. Re David 177/11 - Bari
EVERY WARE COMPUT - V.le Commenda 21 - Brindisi
H & S di Marone M - V. Salomone 56 - Foggia
MONDIAL SOUND - V. Giulio Petroni 48 - Bari
MUSICA UNO SERVICE - V. Cap. Magrone 62 - Molfetta
TECNO UFFICIO SNC - P.zza Giovanni XXIII 10 - Gallipoli

SARDEGNA

AUDIO LINEA SAS - V.le Mameli 60 - Sassari
BAJARDO CARTOLERIA - V. Italia 16 - Sassari
BIT SHOP di Vera Conti e C. - V. Zagabria 47 - Cagliari
CGSI SAS - V. Puccini 4 - Tempio Pausania
COMOS SRL - V. Trieste 57 - Selargius
CSI SRL - V. Satta 8/12 - Carbonia
FRONGIA MARIO - V. Sonnino 94 - Cagliari
IL COMPUTER SNC - V. Ricovero 42 - Oristano
INF. TEL. di Bracci e C. - V. Pergolesi 298 - Cagliari
SAREL di Manca - V. Manzoni 12/14 - Nuoro
SISTEM'S ROOM SNC - P.zza Civica 27 - Alghero
TELE SARDA - V. Roma 62/H - Olbia

SICILIA

AM VIDEO TV - C.so Pisani 312 - Palermo
AP ELETTRONICA - V. Noto 36/38 - Palermo
BENEDETTO RICCARDO - V. Asti 18 - Patti
BIT ELETTRONICA - V. Siracusa 30 - Palermo
BIT INFORMATOCA - V. Gaspare Romano 21 - Mazzara del Vallo
C.H.C. - V. Canfora 122 - Catania
C.H.M. - V. Del Vespro 58 - Messina
CENTRO INFORMATICA 2000 - V. Quieti 7 - Trapani
CINEVISION - V. Palazzi 179 - Gela
COMPUTER SHOP - V.le Orlando 164 - Catania
COMPUTER SOFT CENTER - V. S. Simeone 15 - Siracusa
COMPUTERS SRL - C.so Umberto 81 - Scicli
COMPUTIME - V. Statale 15 - Giammoro
CONDORELLI - V. Renato Imbriani 65 - Catania
DATACOM - V. Pietro Nenni 26 - Agrigento
DONZELLA GIUSEPPE - V. Statale 81 - Ispica
ELECTRONIC di Ciccolò - V. Roma 91 - Barcellona
ELECTRONIC CENTER - V. Renato Imbriani 64 - Catania
FERRAUTO - C.so Umberto 233 - Caltanissetta
GUCCIONE MARIA - V. San Biagio 79 - Comiso
INFRUTTUOSO PASQUALE - V. Vitt. Emanuele 39 - Francofonte
LA CARTOTECNICA - C.so Sicilia 59/61 - San Cataldo
LA MANTIA - C.so Calatafimi 722 - Palermo
MELCHIONDA NAZARIO - V. Simeone 15 - Siracusa
MELLEA SALVATORE - V. Umberto 151 - Augusta
MILICI FRANCESCO - C.so Umberto 24 - Siracusa
NUOVA DIMENSIONE - V. Buganza 11 - Messina
PRESTI GIOVANNI - V. Umberto 162 - Giardini Naxos
TOMA GIOVANNI - V. Novontuno 1 - Castelvetra
UFFICIO MODERNO - V. N. Costa - Pachino

TOSCANA

BOVINI VASCO - V. L.B. Alberti 3 - Arezzo
CAFF Centro HYFY - V. A. Allori 52 - Firenze
CIPOLLA ANTONIO - V. Veneto 26 - Lucca
COMPUTER LINE - V. G. di Vittorio 10 - Firenze
COMPUTER LINE - V. S. Lavagnini 20 - Firenze
ELETT. CENTO STELLE - V. Centostelle 5/B - Firenze
ELETTRIC DREAM SNC - V. Sette Soldi 32 - Prato
ETRURIA FILM di Perinti - Vicolo dello Sportello 13 - Siena
GIMIGNANI ROBERTO - V. Romana 92 - Lucca
I.C.S. SRL - V. Garibaldi 46 - S. Giov. Va.
M.G. di Maurri Gigliola & C. - V. Fratelli 23/29 - Pontassieve
MUSIC RAMA - V. D. Alighieri 17 - Sesto Fiorentino
SABEMA - V. Tosco Romagnola 42 - Empoli
TECNINOVAS COMPUTERS SRL - V. Emilia 36 - Pisa
TELEINFORMATICA TOSCANA - V. Bronzino 36 - Firenze

TRE VENEZIE

ANDRIGHETTI SILVIO - V. Michiel 5 - Pieve di Sacco
APL COMPUTER SRL - V. Tombetta 35/A - Verona
AREM SAS di Poli Ivo & C. - C.so Cavour 35 - Verona
ATRE di Arcangeli & C. - P.le Firenze 23 - Bassano del Grappa
B.B.F. SNC - V. Gramsci 22 - Rovigo
BARBOLIN ELIO - V. Roma 52 - Campo S. Martino
BIT COMPUTER SRL - V. Verdi 8 - Mestre
BONTADI OSCAR - P.zza Verdi 15/B - Bolzano
BRANCALEON F.LLI GABBIA - V. S. Marco 5476 - Venezia
BROLLO ANGELO - V. Alessi 35 - Gemona del Friuli
CAPUTO R. di Caputo & C. - V. S. Marco 5193 - Venezia
CASOTTO ALBERTO - V.le Stazione 116 - Montegrotto Terme
CBL COMPUTER SNC - P.za Mazzini 15 - Belluno
CENTRO SOFTWARE VENETO - V. Colleoni 30 - Thiene
CLINICA DEL RASOIO/COMPUTER - V. Fiume 33 - Rovigo
COMPUTER B. COSTO di Rossi - V. Del Costo 34 - Thiene
COMPUTER LINE - V. C. Battisti 38 - Padova
COMPUTER POINT di D'Andrea - V. Roma 63 - Padova
COMPUTIGI di L. Orel - V.le XX Settembre 55/A - Trieste
CORÒ ALDO - V. Roma 81 - Spinea
ELCOM di Segatti Claudia - C.so Italia 149 - Gorizia
FERCASA SNC - V. Calnova 40 - S. Donà di Piave
FOX ELETTRONICA - V. Maccani 36/5 - Trento
FRANCOMPUTER - C.so Fogazzaro 139 - Vicenza
GOLFETTO GIOVANNI - V. Desman 149 - S. Maria Sala
HOBBY ELETTRONICA di Casale - V. Caboto 24 - Pordenone
HS COMPUTER - V. Cantarane 63/C - Verona
IL GIOCATTOLO 2 - V. Mercato Vecchio 29 - Udine
MAZZUCCATO OTTAVIO - V. G. Galilei 113 - Albignasego
MICROTEC SRL - V. Sarnes 7 - Bressanone
MITHO SRL - V.le Porpetto 11 - Lignano Sabbiadoro
MOFERT SNC di Morvile-Feula - V.le Europa Unita 41 - Udine
MOLON SERGIO - V. F. D'Acquapendente - Padova
PALESA GIORGIO - V. Calmaggiore 10 - Treviso
PARADISO DEL BAMBINO - V. Umberto I 20 - Oderzo
PERSONAL WARE - V. Del Pontiere 2 - Verona
QUAGGIO ACHILLE - V. Veneto 124 - Campolongo Magg.
RADIOFONIA - V. C. Battisti 43 - Cortina D'Ampezzo
RIGO SERGIO - C.so Vittorio Emanuele 33 - Pordenone
RTE di Nicolini & C. - V. Galvani 32 - Valdagno
SAVING COMPUTER SRL - V.le Gramsci - Mirano
SIDE STREET - V. S. D'Acquisto 8 - Montebelluna
TALAMINI LIVIO & C. Sdf - V. Garibaldi 2 - Treviso
TECHNOLOGY COMPUTER HOUSE - Riva Vena 889 - Chioggia
TECNO DELTA Sdf - V. Nordio 9 - Trieste
TECNO POWER COMPUTER SHOP - V. S. Giacomo 30 - Monfalcone
TELMA ELETTRONICA SNC - V. Feltre 244/B - Belluno
TESTI FERRUCCIO SAS - Cà Stimabile 30 - Padova
UP TO DATE di Viel Renzo - V. Vittorio Veneto 43 - Belluno
ZATTARIN GET SRL - V. M. Polo 43 - Mestrino
ZELLA ADELIO - P.zza De Gasperi 31/A - Padova
ZUCCATO SRL - C.so Palladio 78 - Vicenza

UMBRIA

C.S.E. - V. Garibaldi 3 - Terni
COMPUTER HOME - V.le Trento e Trieste 67 - Spoleto
LIBRERIA LA FONTANA - C.so Vannuggi 22 - Perugia
MICROCOGIT - V. Filosofi 24 - Perugia
RAGNI RITA - P.zza XXV Aprile 31 - Umbertide
RASTELLI - V. Baglioni 17 - Perugia
SERLUBINI - V. S. Rocco 22 - Bastia
SUPER ELETTRONICA - V. del Leone 3 - Terni

DISTRIBUTORI

UMBRIA

HARD & SOFT Via Bolzello 2 - 05100 Terni - Tel. 0744-451152

LOMBARDIA

CONSOLO & LONGONI S.P.A. Viale Dell'Industria 63 - 20037 Paderno Dugnano (MI) - Tel. 02-9183372 - Fax 02-9184083
EDELEKTRON Piazza Pattari 2 - 20122 Milano - Tel. 02-800444

PIEMONTE

GRUPPO SISTEMI TORINO Via Reiss Romoli 122/9 - 10126 Torino - Tel. 011-2202651

TRE VENEZIE

INTERSERVICES S.R.L. Via S. Pietro 58/A - 35100 Padova - Tel. 049-655654

TOSCANA

TELEINFORMATICA TOSCANA Via Bronzino 36 - 50142 Firenze - Tel. 055-714884

LAZIO

ALFA LEASING S.R.L. Via Illiria 18 - 00183 Roma - Tel. 06-7597701
APC Via Catalani 23 - 00199 Roma - Tel. 06-8392646 - 9393438
DISITACO - Via Ardia 60 - 00199 Roma - Tel. 06-8440766 - 857607

CAMPANIA

LADA S.R.L. Via F. Imparato 33 - 80146 S.G. Ateduccio (NA)

PUGLIA

R.V.F. S.R.L. Corso Cavour 96 - 70121 Bari - Tel. 080-544651 - 545309

SICILIA

BELCO S.R.L. Via Mariano D'Amelio - 90143 Palermo - Tel. 091-547566
ITALSOFT S.R.L. Via Dott. Palazzolo - 94011 Agira (EN) - Tel. 0935-692560

ASSISTENZA TECNICA

HI-FI OKAY S.R.L. Via Conchetta 4 - 20136 Milano - Tel. 02-8394906
BENATO ALESSANDRO Via F. Zonaro 15 - 35132 Padova - Tel. 049-612508
BERTI RUDI Via Danine 21/c - 40121 Bologna - Tel. 051-442151
GENERAL COMPUTER S.A.S. Via Fieravecchia 26 - 84100 Salerno - Tel. 089-237835
TECNE S.N.C. Via Andrea di Sernia 31 - 80123 Napoli - Tel. 081-7612299
D.C.S. ITALIA S.R.L. Via Arbia 60 - 00199 Roma - Tel. 06-867742
ITALSOFT Via Dott. Palazzolo - Agira (Enna) - Tel. 0935-692560
COMPUTER SERVICES S.N.C. Via Reiss Romoli 122/11 - 10126 Torino - Tel. 011-2202666
ALFA LEASING S.R.L. Via Illiria 18 - 00183 Roma - Tel. 06-7597701
TELEINFORMATICA TOSCANA Piazza Pier Vettori 1 - 50142 Firenze - Tel. 055-227991
HARD & SOFT Via Bolzello 2 - 05100 Terni - Tel. 0744-451152
H & S/C & C di Morone Via Salomone 56 - 71100 Foggia - Tel. 0881-24130
G. SAT di Grassi Zona Industriale Predda N. - 07100 Sassari - Tel. 079-260477
R.V.F. S.R.L. Corso Cavour 96 - 70121 Bari - Tel. 080-545309
MIDEL di Boiani Via Galvani 15 - 16100 Genova Sestri Ponente - Tel. 010-677022
C.H.M. S.R.L. Via del Vespro 58 - 98100 Messina - Tel. 090-719254

AGENTI

EMILIA ROMAGNA

GIANNASI MARCO Via Passo Buole 2 - Reggio Emilia - Tel. 0522-294805

TOSCANA

PRESENTINI GIANCARLO Via Medici 2 - Terranova Braccionini (AR) - Tel. 055-973151

LAZIO

RICCI LUIGI Via F. Inghirani 10 - Roma - Tel. 06-6237040

CAMPANIA E CALABRIA

ERMES Via S. Luca 50 - 80132 Napoli - Tel. 081-402419

LIGURIA

R & R Via F.lli Canepa 94 - 16010 Serra Riccò (GE) - Tel. 010-750729 - 750866 - 752041 - 752051

SARDEGNA

GRASSI GIORGIO Zona Industriale Predda Niedda - 07100 Sassari - Tel. 079-260477

PIEMONTE

MARDUE - Via Monbasilio 62 - 10136 Torino - Tel. 011-3290769 - 328363

TUTTI I PREZZI DEGLI ATARI

CENTRI SPECIALIZZATI DI ASSISTENZA

***HI-FI OKAY S.R.L.**, Via Conchetta 4, 20136 Milano, 02-8394906*

***BENATO ALESSANDRO**, Via F. Zonaro 15, 35132 Padova, 049-612508*

***BERTI RUDI**, Via Dagnine 21/c, 40121 Bologna*

***TECNE S.n.c.**, Via d'Isernia 31, 80132 Napoli, 081-7612299*

***D.C.S. ITALIA S.R.L.**, Via Arbia 60, 00199, Roma*

***ITALSOFT**, Via Dott. Palazzolo, Agira, Enna, 0935-692560*

***COMPUTER SERVICES S.n.c.**, Via Reiss Romoli 122/11, 10126 Torino, 011-2202666*

***ALFA LEASING S.R.L.**, Via Illiria 18, 00100 Roma, 06-7597701*

***TELEINFORMATICA TOSCANA**, P.za Pier Vettori 1, 50142 Firenze, 055-227991*

***HARD & SOFT**, Via Bolzello 2, 05100 Terni, 0744-46658*

***H & S/C & C** di Morone, Via Salomone 56, Foggia, 0881-24130*

***ORE** di Grassi, Zona Industriale Predda N., 07100 Sassari, 079-260477*

***R.V.F. S.R.L.**, Corso Cavour 96, 70121 Bari, 080-545309*

***MIDEL** di Boiani, Via Galvani 15, 16154 Genova, 010-620602*

Categoria	Codice	Descrizione		Prezzo
PERSONAL COMPUTER	**520STm**	Computer 512Kbyte RAM, 192Kbyte ROM, Mouse e Modulatore TV	lire	540.000
	1040STf	Computer 1Mbyte RAM, 192Kbyte ROM, Mouse e floppy doppia faccia 720Kbyte (formattati) incorporato	lire	1.090.000
	SF354	Disk drive 500Kbyte (360Kbyte formattati)	lire	250.000
	SF314	Disk drive 1Mbyte (720Kbyte formattati)	lire	420.000
	SM124/5	Monitor monocromatico alta risoluzione (640×400)	lire	295.000
	SC1424	Monitor a colori RGB	lire	590.000
	SMM804	Stampante a matrice d'aghi 80 caratteri	lire	420.000
	SH204	Hard disk 20Mbyte (formattato)	lire	890.000
	NL-10	Stampante STAR 80 colonne 120 cps NLQ	lire	795.000
	Mouse		lire	95.000
	TOS ROM	Kit ROM per ST	lire	95.000
PERSONAL KIT	**A100**	Kit comprendente 520STm Computer 512Kbyte RAM, 192Kbyte ROM, Mouse e Modulatore TV SF354 Disk drive 500 Kbyte (360 Kbyte formattati)	lire	790.000
	A400	Kit comprendente 1040STf Computer 1 Mbyte RAM , 192Kbyte ROM, Mouse e floppy doppia faccia 720Kbyte (formattati) incorporato SM1424 Monitor monocromatico alta risoluzione	lire	1.290.000
	A450	Kit comprendente 1040STf Computer 1 Mbyte RAM , 192Kbyte ROM, Mouse e floppy doppia faccia 720Kbyte (formattati) incorporato SC1424 Monitor RGB Thomson/Atari	lire	1.540.000
HOME COMPUTER	**130XE**	Computer 128Kbyte RAM, 32Kbyte ROM	lire	199.000
	XC12	Registratore a cassette	lire	59.000
	A1050	Disk drive	lire	199.000
	A1029	Stampante a matrice d'aghi	lire	199.000
	A1027	Stampante di qualità	lire	189.000
	A1020	Stampante Plotter 4 colori	lire	99.000
	CX77	Tavoletta grafica	lire	79.000
	AX130/1	Kit comprendente 130XE computer 128Kbyte RAM, 32Kbyte ROM XC12 Registratore a cassette	lire	249.000
	AX130/2	Kit comprendente 130XE Computer 128Kbyte RAM, 32Kbyte ROM A 1050 Disk drive	lire	359.000
VIDEOGAMES	**JR2600**	Video gioco	lire	84.000
	CX24	Super controller	lire	14.500
	.....	Cartucce ... a partire da lire	lire	9.000

Da oggi Atari gioca duro!

ATARI 1040. Tecnologia forte, prezzo vincente.

Prima non c'era. Adesso c'è.
È il nuovo Atari 1040: la risposta Atari ai Personal Computer che si facevano forti solo del prezzo. Oggi tutti possono avere un computer serio a un prezzo da ridere. Infatti il 1040 vi offre 1 milione di bytes di memoria, il Mouse, un monitor dalla definizione strabiliante (680 x 400 punti), tre generatori di suono e una montagna di programmi che hanno già superato l'esame di milioni di utenti. E tutto questo a molto meno di quanto osereste pensare. Solo Lit. 1.290.000 con monitor in bianco e nero ad alta risoluzione e Lit. 1.540.000 con monitor a colori.* Da oggi Atari gioca duro per vincere. Fate i vostri conti e poi scegliete con chi schierarvi.

*I.V.A. esclusa.

Il computer comprende le seguenti specifiche: 1040 Kb RAM, 192 Kb ROM con sistema operativo, Disk-drive da 720 Kb (formattati). Interfacce: seriale, parallela, mouse, joystick, floppy disk, hard disk, MIDI.

ATARI ITALIA S.p.A. - Via dei Lavoratori, 25
20092 Cinisello Balsamo (MI) - Tel. (02) 6120851 - Tlx 325832

ATARI®

IL COLOSSO ENTRA IN GIOCO

Per richiesta nominativi, agenti e distributori rivolgersi a ATARI ITALIA s.p.a.